Build Your Own **Supercomputer** From Your Macs Laying Around

MacTech Magazine
January • 2006

MACTECH®

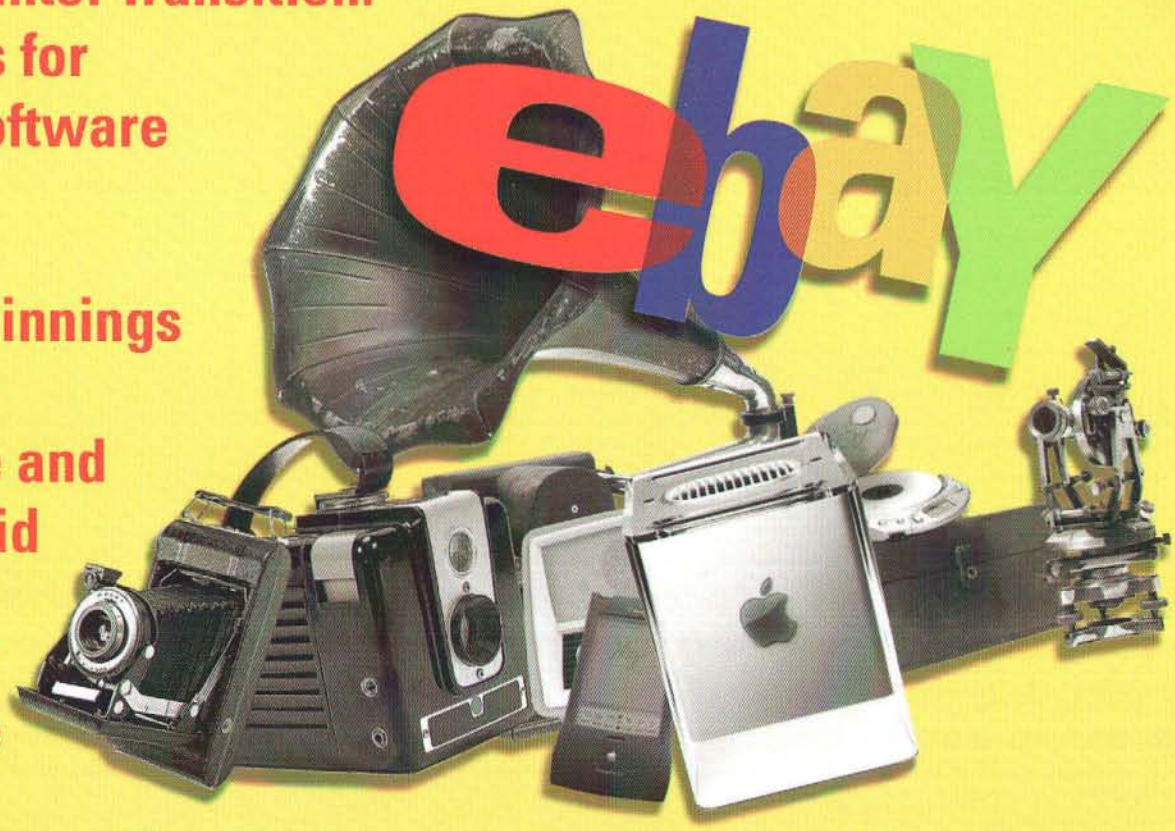
The Journal of Macintosh Technology

Make eBay *Work for You* Instead of You Working for It

The Apple - Intel Transition: Your Options for Windows Software on the Mac

Unix Underpinnings on The Mac: What To Use and What to Avoid

Adding Ajax to a Website



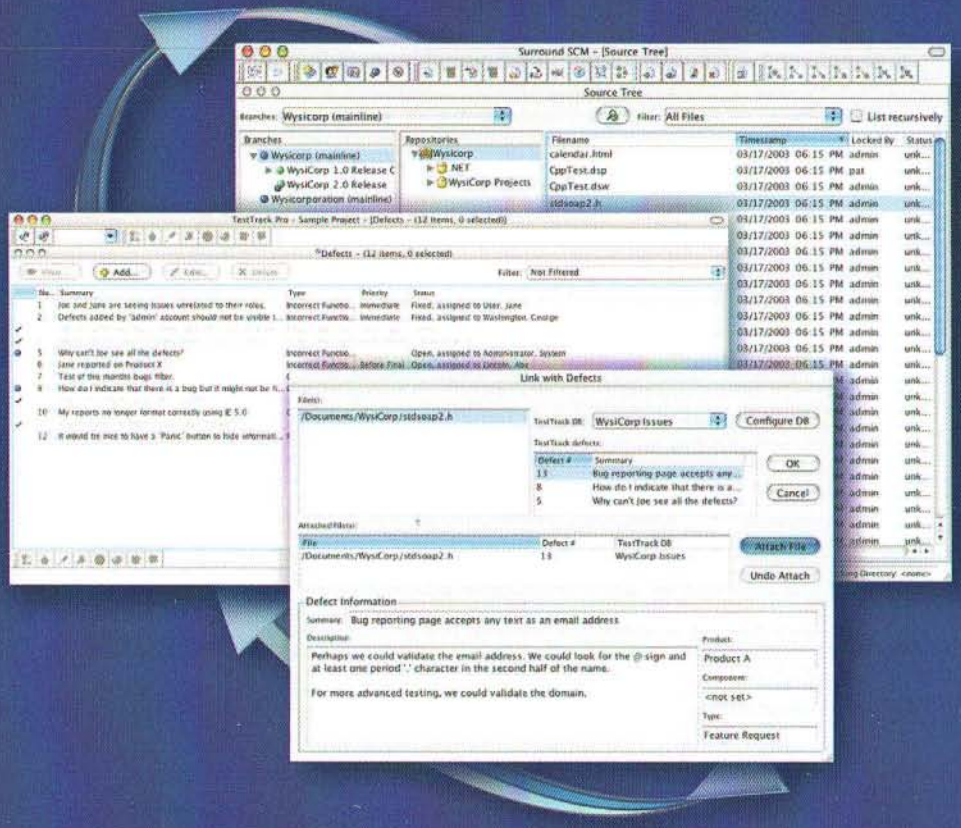
MacTech's Fearless Predictions For 2006!

$8.95 US,
$12.95 Canada
ISSN 1067-8360
Printed in U.S.A.

0 32128 74887 8 01

Complete Source Control and Defect Management for Mac OS X

Seapine Software™
changing the world of software development



Effective source code control and defect tracking require powerful, flexible, and easy-to-use tools—Surround SCM and TestTrack Pro

- Complete source code control with private workspaces, automatic merging, role-based security, and more
- Comprehensive defect management — track bug reports and change requests, define workflow, customize fields
- **New!** Full Unicode support maintains international characters across various languages, ensuring data is not lost or misinterpreted
- Advanced branching simplifies managing multiple versions of your products
- Fast and secure remote access to your source files and defects — work from anywhere
- Scalable and reliable cross-platform, client/server solutions support Mac OS X, Windows, Linux, and Solaris
- Exchange data using XML and ODBC, extend and automate with SOAP support
- Licenses priced to fit your budget

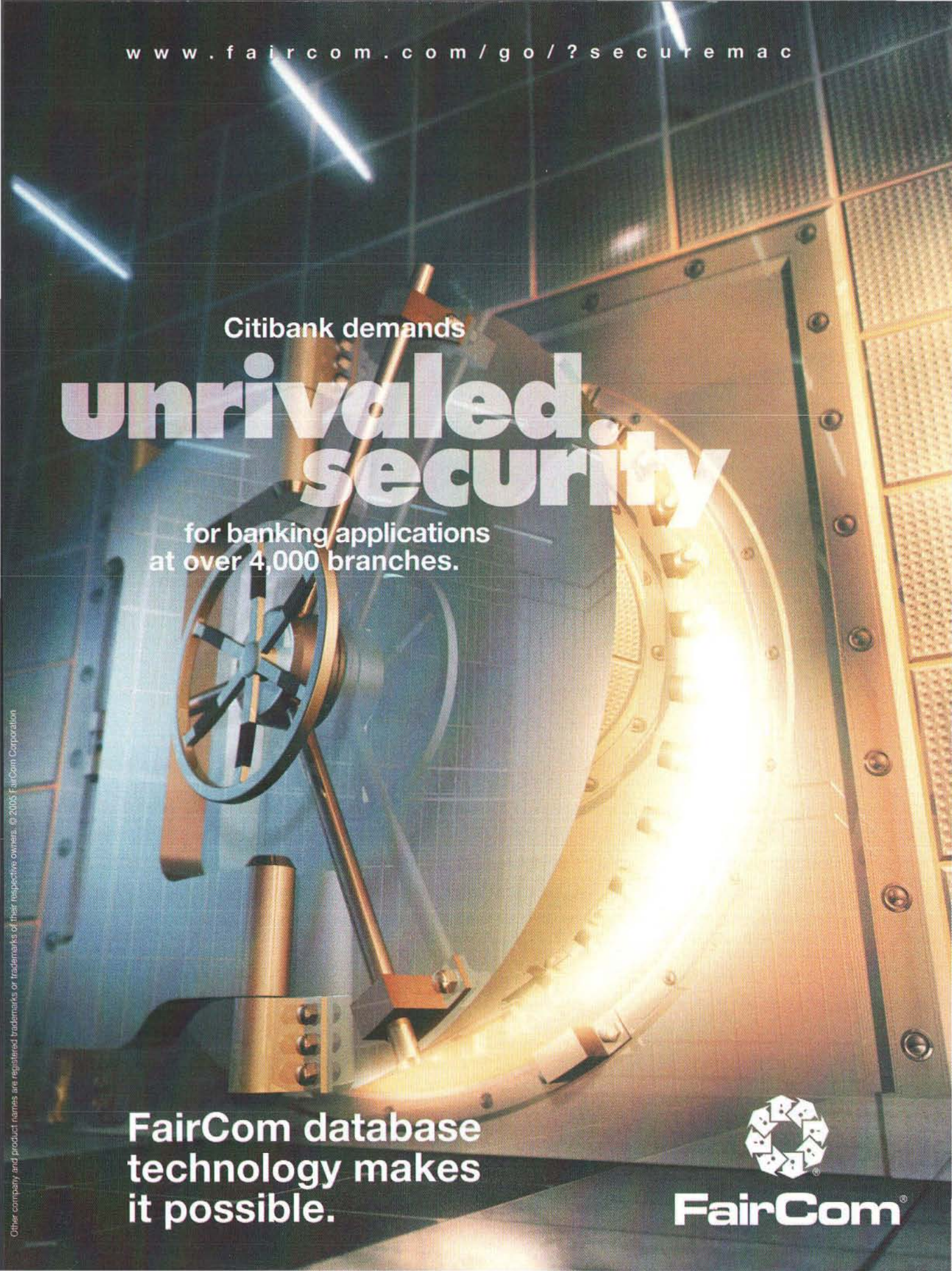
Seapine Software Product Lifecycle Management
Award winning, easy-to-use software development tools

Seapine Surround SCM
Seapine TestTrack PRO

software development 13th annual product excellence award JOLT

Download Surround SCM and TestTrack Pro at
www.seapine.com/mac
or call 1-888-683-6456

all product names listed herein are registered trademarks of their respective owners. All rights reserved.


www.faircom.com/go/?securemac
Citibank demands
unrivaled security
for banking applications
at over 4,000 branches.
FairCom database
technology makes
it possible.
FairCom
Other company and product names are registered trademarks or trademarks of their respective owners. © 2005 FairCom Corporation

Text Recognition Pen Scanner

IRISPen express

You scan. It types!

Instantly enters text and numbers into your computer

Extremely fast and incredibly accurate, the IRISPen Express is the most effective solution for your retyping needs!

It instantly encodes text and numbers into ANY application - such as word processors, e-mail software, on-line databases, spreadsheets, accounting software, etc. What you get is real editable text, right at the cursor position!

Save time and even enjoy your usual tedious data-entry!

Fun and fast, the IRISPen Express works just like a highlighter. Slide the pen over printed text and numbers, they are automatically retyped at up to 1.000 characters per second!

Smart, Portable, Versatile

The IRISPen Express is light, compact and portable. It installs in 3 min. Use it with any laptop and desktop, wherever you go, wherever you are!

MacTech Special

$99 instead of $129, order today at www.irisusa.com/mactech or call 1 800 447 4744

www.irisusa.com
info@irisusa.com

I.R.I.S.
Document to Knowledge™

TABLE OF CONTENTS

ARTICLES & DEPARTMENTS

MacTech's Fearless Predictions for '06...and Beyond!!
by MacTech Staff 6

The Source Hound
The Apple - Intel Transition: Your Options for Windows Software on the Mac
by Dean Shavit 8

Build Your Own Supercomputer From Your Macs Laying Around
by Mary Norbury-Glaser 16

AppleScript Essentials
Providing Progress Feedback During Script Execution
by Benjamin S. Waldie 26

Make eBay Work for You Instead of You Working for It
by Neil Ticktin 38

Mac In The Shell
Unix Underpinnings on The Mac: What To Use and What to Avoid
by Edward Marczak 42

Adding Ajax to a Website
by Andrew Turner 48

Choosing A Java Scripting Language
by Paul T. Ammann 60

Lasso Professional Server 8

the fastest, easiest way to tie any database to any website on any platform.

Lasso Professional – the next generation of building data-driven Web sites – is simply the fastest, easiest way to tie **your** databases to the Web. It supports multiple databases simultaneously, eases the sequential migration of databases and it's fully supported on Mac, Windows® and Linux.

4 FREE well commented Lasso solutions to get you started:

Blog: Teaches beginners and advanced users alike how to create an online diary, or web log.

QuickPoll: Add a poll to your Web site with a question and answers, cast a vote or see results.

Collaborate: Enables knowledge sharing among a group of developers and administrators to streamline Web site development from remote locations or geographically dispersed teams.

Message Board: Source code for the Message Board solution is completely open and expandable to suit your needs.

Special Offer
for MacTech Readers

Try Lasso Professional Server 8 FREE for 30 days.

www.TryLasso.com
lassosales@omnipilot.com
800-678-9958

Then get 60 days of free support with your purchase.

OMNI PILOT SOFTWARE

© 2006 OmniPilot Software, Inc.

Lasso

Communicate With Us

Department E-Mails

Orders, Circulation, & Customer Service
cust_service@mactech.com

Press Releases
press_releases@mactech.com

Ad Sales
adsales@mactech.com

Editorial
editorial@mactech.com
(Authors only, no pr)

Accounting
accounting@mactech.com

Marketing
marketing@mactech.com

General
info@mactech.com

Web Site
http://www.mactech.com

In this electronic age, the art of communication has become both easier and more complicated. Is it any surprise that we prefer **e-mail?**

If you have any questions, feel free to call us at 805/494-9797 or fax us at 805/494-9798.

If you would like a subscription or need customer service, feel free to contact MacTech Magazine Customer Service at 877-MACTECH

We love to hear from you! Please feel free to contact us with any suggestions or questions at any time.

Write to letters@mactech.com or editorial@mactech.com as appropriate.

MACTECH®
The Journal of Macintosh Technology

A publication of XPLAIN CORPORATION

The Editorial Staff

Publisher & Editor-in-Chief: Neil Ticktin
Associate Publisher & Editor: David Sobsey
Editor-at-Large: Dave Mark
Production: Dennis Bower

Xplain Corporation Senior Staff

Chief Executive Officer: Neil Ticktin
President: Andrea J. Sniderman
Accounting: Marcie Moriarty
Customer Relations: Susan Pomrantz
Board of Advisors: Steven Geller, Alan Carsrud

Regular Columnists

QuickTime ToolKit: by Tim Monroe
Patch Panel: by John C. Welch
The Source Hound: by Dean Shavit
Reviews/KoolTools: by Michael R. Harvey
AppleScript Essentials: by Ben Waldie
Mac In The Shell: by Ed Marczak

Board of Advisors

Chairman: Dave Mark
Jordan Dea-Mattson, Steven Geller,
Bruce Friedman, and Richard Kimes

Contributing Editors

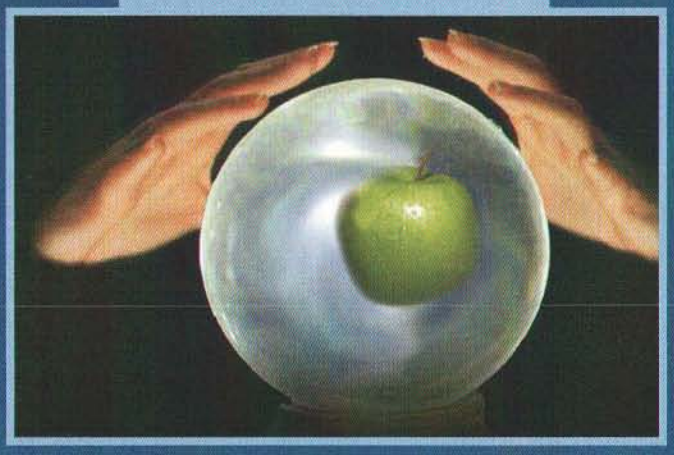
Michael Brian Bentley, Gordon Garb, Vicki Brown, Chris Kilbourn
Marshall Clow, Rich Morin, John C. Daub, Will Porter, Tom Djajadiningrat
Avi Rappoport, Andrew S. Downs, Cal Simone, Steve Sisak

MacTech Magazine (ISSN: 1067-8360 / USPS: 010-227) is published monthly by Xplain Corporation, 850-P Hampshire Road, Westlake Village, CA 91361-2800. Voice: 805/494-9797, FAX: 805/494-9798. Domestic subscription rates are $47.00 per year. Canadian subscriptions are $59.00 per year. All other international subscriptions are $97.00 per year. Domestic source code disk subscriptions are $77 per year. All international disk subscriptions are $97.00 a year. Please remit in U.S. funds only. Periodical postage is paid at Thousand Oaks, CA and at additional mailing office.

POSTMASTER: Send address changes to **MacTech Magazine**, P.O. Box 5200, Westlake Village, CA 91359-5200.

All contents are Copyright 1984-2006 by Xplain Corporation. All rights reserved. MacTech and Developer Depot are registered trademarks of Xplain Corporation. RadGad, Useful Gifts and Gadgets, Xplain, DevDepot, Depot, The Depot, Depot Store, Video Depot, Movie Depot, Palm Depot, Game Depot, Flashlight Depot, Explain It, MacDev-1, THINK Reference, NetProfessional, NetProLive, JavaTech, WebTech, BeTech, LinuxTech, MacTech Central and the MacTutorMan are trademarks or service marks of Xplain Corporation. Sprocket is a registered trademark of eSprocket Corporation. Other trademarks and copyrights appearing in this printing or software remain the property of their respective holders.

MacTech's Fearless Predictions for '06...and Beyond!!



We thought it would be fun to poll our team of Mac aficionados and see what they're thinking regarding the near term future of the Mac marketplace. So we asked our team of editors, contributors and friends what they thought would be the most important Mac marketplace developments in 2006. In no particular order, here's what we are looking for in the next 12 months. Hey, we know we're going to be wrong, but what the heck. This is always fun to do, if for no other reason, just to see how wrong we are!

It's an iPod World!

Not surprisingly, many of our predictions involve this amazingly successful area of Apple's business. Here are a few we found particularly interesting:

Dave Mark

- I can't wait to see what Steve Jobs has up his sleeve for the iPod in 2006. My bet is that we'll see a slow trickle of entertainment deals for the iTunes Video Store. The big question is, will outfits like Sony be willing to fork over their considerable intellectual property to a company they consider a competitor in many respects. At the very least, I expect the market for DVD->iPod Video ripping software to continue to evolve. Question is, will the iPod ever support an external cartridge format, a la the PSP UMD, so I can bring a baggie full of movies with me on the plane?
- My long shot prediction for 2006 is a GPS capability for the iPod. How cool would it be if you could use your iPod to find your way from your hotel to that new Sushi restaurant you read about? Of course, I'd want to equip mine with a depth/fish finder and permanently mount it on my boat, but that's another story

David Swain

- In an effort to get in on the corporate "be seen with iPod" craze, Budweiser is planning on introducing a beer hat that holds two cans on the sides and has a pocket for an iPod over the visor. There are still camps within the marketing group championing either "earBud" or "iBud" for the name of the device, but it is sure to be a hit come Spring Break 2006. The cap also sports a small remote attached to the main siphon tube for controlling the iPod with minimal effort...

Paul Ammann

- My prediction is that Apple will equip all their iPods with On*Star to provide safety and security to help protect users while listening to their iPods. It would be cool if Apple could monitor your iPod's performance and alert you: "Hello, Mr. Sobsey? We've been alerted that your iPod is starting to experience a minor performance issue. Would you like me to make an appointment for you?"
- Additionally, Apple will also team up with Life Alert and offer this service to iPod users. This would be great for everyone, especially if someone has a medical condition and needs emergency services.

- Other possibilities would be to have LoJack, so if your iPod is stolen police could track it down. And since RIM is having a terrible time in court regarding its Blackberry (as of this writing), iPods will replace Blackberries throughout the US for all IT professionals and management.

Ed Marczak

- iPod: The iPod will....ummmm....get better!

Intel Based Macs

Another area people are very interested in is the impact Macs carrying Intel CPUs on their motherboards

will have in the market. Here's a few very interesting predictions on this important area.

Emmanuel Stein

- Soon after the release of x86-based Macs, the Darwine project will enable Intel-based Macs to run Windows applications at near-native speeds
- Apple will implement Linux binary compatibility in OS X on x86, as Sun did with Solaris, thereby enabling all those cool ELF-based binaries to run on OS X without porting and/or re-compilation.
- Doom 3 on OS X for Intel will FINALLY run as fast as it does on Windows
- Under increased pressure to differentiate themselves from commodity x86 vendors, Apple will expand and innovate their hardware offerings to appeal to key market segments such as Gamers, Creative Professionals, and Scientists with multi-segment products such as Apple's version of an affordable stereoscopic 3D visualization goggles/glasses
- The top of the line Macs for 2006 will be have at least 4 dual-core processors x86 processors by the end of year
- Apple will initially be frustrated by its relative inexperience with enterprising hackers on the x86 platform who will be relentless in their efforts to hack Mac OS X to run on commodity x86 hardware as has been the case (I hear) with pre-release versions of Tiger x86. By year's end Apple will have addressed the issue, possibly working with Intel, to develop increasingly complex and irreversible hardware and software techniques to prevent pirating of OS X on standard x86 boxes.

Dave Mark

- One of the biggest product launches for 2006 will be the first run of Intel-based Macs. I'll be very interested in seeing if the lag between desktop to laptop is less with Intel than with IBM/Moto, where we still haven't seen a G5 laptop.

Ed Marczak

- The Intel Macs will go off without a hitch. Transition will be as seamless as 0x0 to PPC.
- The PPC Macs: PPC owners will gripe as the PPC OS builds fall behind the x86 version from time to time.

Generally Interesting Stuff

To wrap things up, here are some interesting (and in some cases humerous!) thoughts that really couldn't be easily categorized, but we still thought it would be fun to share these with our readers!

Aaron Hillegass

- In the past, the squirmishes around competing technologies have been numerous but small: emacs vs. vi, ruby vs. python, Firefox vs. Safari, etc. (This is because while everyone cares about some battle, only a small minority of people care about any particular battle.) This will change radically in 2006 when the "Stack Wars" break out. By creating (and naming) a stack of technologies, you can create a large group of people who will argue rabidly in favor of emacs/Python/PostgreSQL/Linux/Gtk+. These people will have distain for another large group who will argue just as fervently for Xcode/Ruby/MySQL/Mac/Cocoa. By the end of 2008, there will only be one argument between two stacks, and everyone will be on one side or the other. The argument will end abruptly when we run out of electricity. There will be a brief silence, and then someone will yell, "You're an idiot; the abacus sucks! Look at what I can do with my slide rule!"

Emmanuel Stein

- Apple's Xserve line will continue to exploit the raw computing power of IBM's Power architecture, whose FPU performance has made it a platform of choice for cost-conscious (and not so cost conscious) large scale computational clusters. Apple will extend their cocoa Xgrid API, allowing developers greater flexibility and potential in developing distributed computation solutions for the Creative Professional market. Expect initial offerings to follow the Seti@Home models using screen-savers to render computationally expensive task such as rendering of video, followed by increasingly complex models, which depend on tightly coupled inter-node communication for effective processing and that take advantage of the increased presence of low latency network interconnects.

MacTech Staff

- Apple will assume for video the preeminent spot it now has for audio. It will enter and dominate the set-top box market
- As a result, Apple stock will pass the $150 mark by 31 December 2006
- Apple Computer will buy Apple Records and put an end to the lawsuits
- In overwhelmingly positive response to the incredibly valuable content found each and every month in the "new" MacTech, our subscriber base will continue to grow and increase by at least 50%

See you next year!!

MT

THE APPLE – INTEL TRANSITION

YOUR OPTIONS FOR WINDOWS SOFTWARE ON THE MAC...

When Steve Jobs walked out onto the stage to deliver his keynote at Apple's World Wide Developer Conference (WWDC) this past June and revealed to the largest gathering of Apple Developers in the twenty-one year history of the Macintosh platform that Apple was going to transition its entire product line to Intel processors, there was a mixture of excitement and stunned silence in the hall.

Game Over: The Prodigal Son Has His Way

Most everyone knew that such a transition was squarely within the realm of technical possibility, as OpenStep, the immediate predecessor to Mac OS X had been a product for Intel processors, but the likelihood of such a major shift in the personal computing landscape, although rumored in the mainstream media prior to WWDC (and a good thing too, as I think the Moscone West convention center probably didn't have enough portable defibrillators handy for a **complete** surprise) wasn't really what the audience was expecting. Over the course of the last six months, a shockingly small amount of analysis of the event has appeared in the media, as if no one seems to be able or willing to articulate the subtexts of what the transition means. Now, as MacWorld San Francisco 2006 and another keynote address approaches, journalists and Mac professionals are beginning to see the signs of a sooner-rather-than later MacIntel release. Indeed, by the time you read this article, you may be typing in your pre-order for the new Macintosh, if that's what it's called.

Goodbye, Old Friend

At WWDC 2002, Steve Jobs ceremoniously laid Mac OS 9 to rest and proclaimed it "dead." At WWDC 2005, Steve Jobs, in so many words, proclaimed the Macintosh "dead," without actually saying so. Hold on. Before you take this magazine and run it through the shredder, with my photo leading the way, ask yourself the following: "What makes a Macintosh different from all other personal computers?" The first words to your lips are probably "The operating system." Right. The second thing to your lips "The hardware." Right again. We already know that aside from a few "special" components, like Intel's Trusted Computing Initiative chips, there isn't going to be much that separates a Macintosh from any other PC except for the Apple Logo on the outside of the box and Mac OS X, its operating system, and nothing to stop people from installing Windows XP on it, either.

For many years, Apple's marketing team chased the windmill of the "Megahertz Myth," to differentiate its computer products from the Wintel oligarchy when it reality it was primarily the software delivered with the hardware that made the Macintosh experience, and not so much the box it ran on. The G5 processor, as great as it is, was adapted from big-iron server hardware, not engineered for a personal computer without requiring an equally impressive cooling system, certainly not suitable for a portable. The roadmap that mattered to Apple, a gigahertz rating that kept pace with Intel's chips and power consumption fit for a notebook, was **not** the roadmap that mattered to IBM.

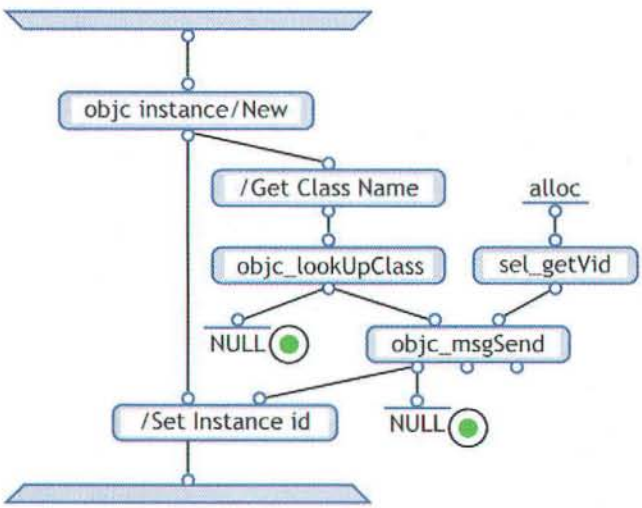
So, the question remains, does Apple want the

Macintosh to die? Let's have a look at the evidence. First, Mac OS 9, the tie that bound the Macintosh to the original product of 1984, was proclaimed dead in 2002. Second, the PowerPC-based Macintosh was declared an endangered species at WWDC 2005, with a gradual phase-in of Intel-based hardware in 2006 with a complete transition finished by 2007. The keynote was impressive, Steve Jobs used an Intel-based prototype for his demo, and in testing the prototypes made available to us during the conference, we all had to admit that they were very, very fast, as fast perhaps as a lower-end G5 tower, without the panoply of fans. Oh, and a little-mentioned fact easy to overlook about the new Intel prototypes, what with all of the talk about Rosetta, the emulation technology that allowed existing Mac OS X applications compiled for PowerPC hardware to run without modification, or Universal Binaries, the "fat" applications developers could produce to run on both processors with Xcode 2.1, was that Classic wasn't going to be supported at all on the new Intel-based computers. Mac OS X Applications not updated as Universal Binaries are going to become the new Classic. The old Classic is destined to become a Fossil relegated to those who buy computers with PowerPC processors. The word is out: **let go of Classic completely,** not just booting in OS 9, **but even the software.**

Figure 1. Apple's Rosetta Emulator

Of course there will be some open-source or third-party solutions that will allow for running Classic, but it's going to be really slow, like running Windows programs in an Intel X86 emulator on current Macintosh hardware, whereas Classic on PowerPC Macs is actually very fast, considering it is a virtual application environment inside of Mac OS X, with a 68000 to PPC translator on the logic board that allows the execution of the pre-Classic legacy software: non-native PowerPC

Draw code
You know you want to
Marten™ IDE



• Edit while you run • One click Cocoa and Carbon application creation • Powerful debugger

w w w . a n d e s c o t i a . c o m

applications. With no critical enterprise or educational software solutions holding OS 9 users back anymore, there's no real reason for Apple to keep its own support staff up to speed on the Classic Mac OS. Hardware manufacturers have already stopped making drivers for OS 9 or the Classic Environment.

So with the ties to the legacy hardware cut, or at least with a path through the transition for those who might not want to upgrade all of their recently purchased Mac OS X Applications made possible by the Rosetta instruction translator, and the ties to the legacy Operating System cut, what is left of the Macintosh with the smiling face that we knew so well? Not much. But a quick glace at Macintosh history shows that the Classic Macintosh hardware (and certainly not the PowerPC) as well as the Classic Mac OS were **not** Steve Jobs' creations, they were the children of other long-gone Apple leaders. But what we will have left after the transition will be a pure creation of Steve Jobs' (a.k.a the ultimate perfectionist) vision for the NeXT computer, and it's going to be better, faster, less expensive, more compatible with a lot of surprising things, and a Macintosh in name only, if, and only if, Apple chooses to continue using that trademark.

I Shall Disown the Child of My Past, and Kidnap Yours

Do you hate yourself? You might have made some regrettable mistakes along throughout your life or career, but how did your mistakes affect the fabric of our technological culture? With the removal of all ties to the Classic Macintosh hardware and software, and with Apple's support for Mac OS 9 software in sight (approximately two to three years away), it's pretty clear that Apple will be able to move Mac OS X in any direction is desires, and cut its spendthrift son of legacy software and hardware off once and for all. The hardware Apple sells will no longer be "special" and "different under the hood," and no longer be a millstone for the marketing folks to attempt to overcome, either. Apple commercials may actually end with the "Intel inside" chimes! Apple computers and servers won't ever have the edge in speed again, or fall into the "gigahertz gap." I've always thought that the so-called "rivalry" between Microsoft and Apple has always been mostly marketing B.S., the reality being that Apple lost the Enterprise market to the Wintel oligarchy a very long time ago. But when Intel-based Apple computers start hitting the streets in 2006, it's truly going to be Apple v. Microsoft for the **very first time,** especially if a big Wintel fabricator, like Dell, gets serious about wanting give their customers a choice. So, does Apple really have a chance to win back the personal computing mainstream or the server room?

Already, there's been a flurry of discussion on several listservs that I frequent about whether dual-booting the new Apple Intel-based computers between Windows XP Professional (and soon Windows Vista) and Mac OS X 10.4 will be a "solution" that Enterprise customers should consider if they want to go to Mac OS X and keep some critical Windows applications in service. I say **are you kidding?** to dual-booting. Nobody likes it. Dual-boot solutions for transitioning users from Windows 3.1 to Windows NT 4 lasted a mere millisecond or two before IT departments and end users alike gave it up. The dual-boot solution for early adopters of Mac OS X 10.0 lasted until 10.1.2 was available. Nobody wants to reboot, not even once a month! It's hard enough for folks to keep track of what's on one file system, let alone two!

The Good Son

Although Microsoft might have made many an IT Manager angry for its rather monopolistic licensing policies that require companies to pay subscription charges for their server and application licenses, as well as CALs (client access licenses), Microsoft has certainly recognized that its own software legacy is what keeps it firmly entrenched in the Enterprise, and while it is known to yank the carpet of OS support out from under users (as it did with NT and Windows 3.1 95, 98 and ME), what Microsoft **hasn't** done is kill support for legacy applications. Good 'ol DOS applications, Windows 16-bit applications, and applications from nearly every rapid-development IDE (integrated development environment) that's important to Windows Enterprise customers still work serviceably in Windows XP Professional and Windows 2000, and although Microsoft occasionally makes some noise about wanting to discontinue support for all those legacy products, it knows where its bread is buttered.

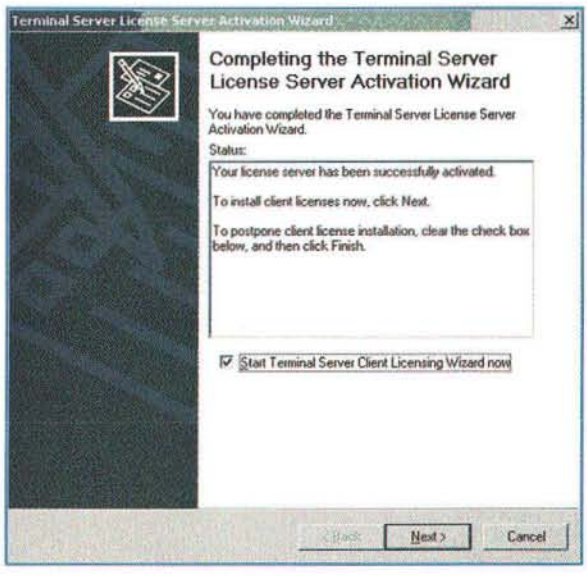


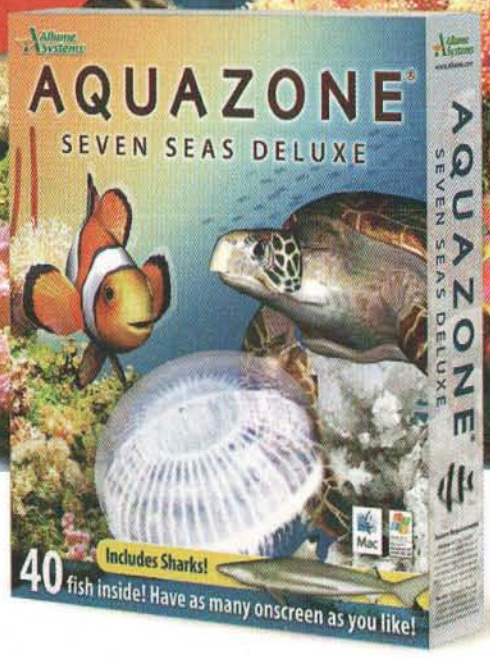
Figure 2. An IT Manager's Favorite Piece of Software—Microsoft Licensing

And Apple knows where Microsoft's bread is buttered as well. A company with an open-minded CTO who is interested in implementing open-source and the quality and security of

YOUR ULTIMATE 3D AQUARIUM!

NEW RELEASE!

**FORTY BEAUTIFUL FISHES,
TURTLES, JELLIES, & SHARKS!
INTERACTIVE SCREENSAVER
TAP THE GLASS, FEED THE FISH**



Allume Systems
A Division of Smith Micro Software

A virtual aquarium so vibrant, you'll swear the fish are real! Available from your favorite retailers and catalogs.

Aquazone Seven Seas Deluxe includes 40 fresh and saltwater fish, jellyfish, adult and baby sea turtles, and sharks! Have entire schools of fish on screen swimming through 20 customizable 3D tanks!

®, TM, and © 2005 Allume Systems, Inc. To learn more about our range of software solutions for your Macintosh, please visit us at **www.allume.com.**

Mac OS X on Apple hardware may realize that deploying XServe G5s would save his company a bundle in annual CAL payments, but cannot even begin to consider putting a Mac OS X computer in his employee's cubicles until a critical database front end, created in some Windows-specific IDE like Powerbuilder will run under Mac OS X. Virtual PC on current Macintosh hardware would be a slow solution at best, hardly worthwhile given the extra expense.

Don't Talk to Strangers

Microsoft has been really great about making sure its good son (legacy software) has the full support (financial and technical) of its rich papa, but some open-source wizards figured out quite some time ago that Microsoft's legacy software is easily tricked into running away with strange operating systems such as...

In Vino Vertias (In Wine There Is Truth)

Linux developers once faced the same issue Apple now faces: how to get a foothold on the slippery slope of the Microsoft-Dominated Enterprise desktop market and make those one or two critical applications work on their operating system. So naturally, an open-source project (that Linux hands are very familiar with) was born: Wine. Located at http://www.winehq.com, the Wine project describes itself this way:

> Wine is an Open Source implementation of the Windows API on top of X and Unix.
>
> Think of Wine as a compatibility layer for running Windows programs. Wine does not require Microsoft Windows, as it is a completely free alternative implementation of the Windows API consisting of 100% non-Microsoft code, however Wine can optionally use native Windows DLLs if they are available. Wine provides both a development toolkit for porting Windows source code to Unix as well as a program loader, allowing many unmodified Windows programs to run on x86-based Unixes, including Linux, FreeBSD, and Solaris.

Wine's been around since 1993, nearly since the beginning of Linux itself, which goes to show how important those few Windows applications can be. It's only a matter of time before Mac OS X for Intel is added to the list of fully supported operating systems.

Instant Lottery Winners

OK, let's say for a moment that Apple is successful with its Mac OS X for Intel transition and in the process gains a modest amount of market share over the next two or three years. Whose prospects will be on the rise, and whose will left on the roadside of progress?

Winners: the Darwine Project and CrossOver Office

While Wine is a great solution for running Windows applications compiled for Intel processors under Linux, a group of ambitious open-source hackers already managed to port Wine to Mac OS X for PowerPC (Darwin). Named Darwine (http://darwine.opendarwin.org) of course, the project was hamstrung by the need to use an emulator to translate X86 Intel instructions to the PowerPC instructions needed by current Power Macs. Now, with Mac OS X for X86 coming, Darwine is ready, waiting, and working. Unfortunately, Apple's Developer NDA precludes me from speaking about the specifics of the pre-release versions of Apple Intel hardware or Mac OS X, but a few searches around the web indicate that a few hackers running Mac OS X for Intel on non-Apple hardware have found Darwine working quite well. When Intel-based Apple computers appear on retail shelves, there's little doubt that Darwine will be the darling that garners the most press, and hopefully the most success stories.

Figure 3. Darwine

Also waiting in the wings is CrossOver Office, a commercial version of Wine with Enterprise support, promising the ability to run the full Windows versions of applications like Photoshop, Microsoft Office 2003, Lotus Notes, Visio Professional, Microsoft Project and other mission-critical programs that previously required Virutal PC to run on Mac OS X. If CrossOver Office can allow an Mac OS X user to run an in-house database application developed in Microsoft Access, for example, that opens the door for a Windows-less, secure, and virus-free Enterprise workstation.

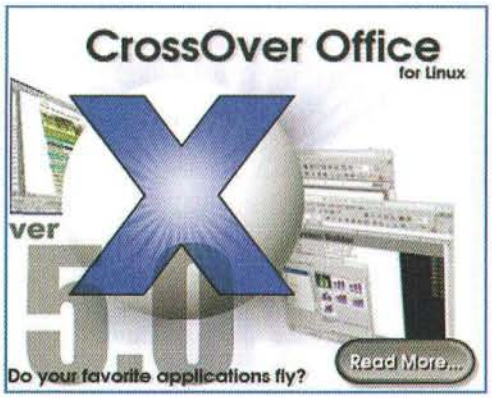


Figure 4. CrossOver Office

The only thing that is standing between Darwine and CrossOver Office lighting the Enterprise world on fire with Mac OS X for Intel is X Windows, currently a requirement for both. If either takes on the challenge of getting Windows applications running in an Aqua window, and well, it would be well-poised for success.

Winners: The Majority of Open-Source Projects for Mac OS X

While Mac OS X currently has a plethora of open-source goodies available for it via Fink (http://fink.sourceforge.net) and Darwinports (http://darwinports.opendarwin.org), many of the supporting libraries lag somewhat behind their Linux and BSD counterparts. Mac OS X on Intel will bring Darwin closer to the open-source development mainstream, rather than as a branch that typically lags behind the Linuxes and BSDs.

Winners: The Virtual Machine Makers, Microsoft, VMWare, and possibly PearPC

If Microsoft meets Apple at the point of Enterprise adoption, there's a chance that the Windows version of Virtual PC could be modified to run transparently in the Mac OS X Finder and become the standard compatibility tool for Mac OS X Intel computers rather than CrossOver Office or Wine. A fairly comprehensive list of what works and what doesn't work in Virtual PC 2004 for Windows is available here: http://vpc.visualwin.com.

On a lower level, VMware virtual Machine software is poised to allow Apple Intel Mac OS X computers to run multiple OSes on the same piece of hardware. Unlike dual-boot solutions, EMC^2 VMware is a multiple-simultaneous-boot solution that literally would allow an Intel Xserve to run Mac OS X Server and Windows 2003 Server on the **same server**, sharing a storage subsystem, like an Xserve RAID and even the same network card. Such options would enable a dreamy upgrade path for IT Managers seeking to save money on file and print services, without having to sacrifice the collaboration tools of Microsoft Exchange, or retooling an Active Directory deployment. While Mac OS X and Mac OS X Server already interoperate smoothly with Active Directory, VMware will change the way we define **interoperate** forever.

Figure 5. VMWare

With its reason for existence (to run Mac OS X on Intel hardware) pretty much snuffed out, the once wildly popular PearPC (http://pearpc.net) project is poised to transform itself into the VMware of the open-source world as a virtualization tool for running multiple instances of X86 operating systems on a single computer. While it has quite a way to go before it can rival the proven functionality of VMware, I'm hoping that the PearPC

JobCapture

Automatic Time and Expense Tracking Software

JobCapture is intuitive, user-friendly client-server software that intelligently and automatically tracks working time and activity. Open a document in any application, and it is automatically being timed by JobCapture. This guarantees that *all work activity is recorded*.

- Perfect for ad agencies, design studios, architects, law firms, PR Firms etc.
- Scalable. Flexible.
- Easy to configure.
- Cost effective. Adding additional users is easy and inexpensive.
- Tracks time and expenses.
- Client and Server software is included.
- Easily customized.
- Real-time data delivery.
- Customizable reporting.
- View data numerically or graphically.
- Import/Export data to SQL, mySQL, XML etc.

JobCapture will *increase productivity and your bottom line* by providing accurate records for billing and analysis. *Stop losing money now—Call today* for more information: 973.763.9494. Reps are available 10:00am-5:30pm, EST.

www.captureworks.com

CaptureWorks

developers respond to the challenge that Apple's hardware shift proposes, and that there's an open-source alternative to VMware available.

Figure 6. PearPC

Winners: Mac IT Professionals with Windows Knowledge and Experience

With the inevitable market share expansion that Mac OS X on Intel will bring in both desktop and servers comes the opportunity for Mac IT Professionals to get involved in larger upgrade projects that may see Windows Enterprise solutions co-existing on Apple Hardware, and deployment projects that bring mission-critical database applications onto the desktop of computers running Mac OS X for Intel by hook by crook or by compiler, and a need to support them as well as a road map to eventually transition such applications to rapid development environments not tied to the Win32 API.

I expect that Mac OS X for Intel will spawn a new generation of addicts and a new class of professional, let's call them Winos—who will make their living creating a safe home for Microsoft's legacy on Apple hardware and OS. Even if Apple fails to reclaim any Enterprise market share with its new hardware, which is unlikely, if just out of curiosity-driven sales, the Intel transition can do nothing but good for Mac OS X users who will have the possibility to use the best products of Microsoft and Apple, whether they know it or not. When the transition to Intel is over in two years, the ties that connected us to the original Macintosh of 1984 will finally be cut, and we'll move forward, faster than we ever have.

MM

About The Author

Dean Shavit is an ACSA (Apple Certified System Administrator) who loves Open-Source and freeware solutions for Mac OS X. During the day, he's a partner at MOST Training & Consulting in Chicago, where he trains system administrators in Mac OS X and Mac OS X Server, helping his customers get the best ROI possible from their computer investment while writing for his own website, www.theMachelpdesk.com. Recently, he became the father of an application: the Mac HelpMate troubleshooting tool, available at www.Machelpmate.com. If you have questions or comments you can contact him: dean@Macworkshops.com.

BIG SECURITY
FOR THE SMALL BUSINESS

MacWorld **"Best of Show"** Winner, **CRYPTOCard** introduces **5-User Kit for OS X**, the "All-In-One-Box" Authentication Solution that lets your security grow with you.

CRYPTO-Server 6.3 is the Mac-centric authentication solution for IT infrastructures organized around Mac OS X Server. Once implemented, your Mac OS X Server becomes the centralized authentication and token management system for all users, regardless of their computing platform or location.

FOR MAC DESKTOPS

- Enforced **CRYPTOCard** authentication for desktop access
  – *Startup, Screen Lock, Sleep*
- Authenticated Fast User Switching
- Internet Connect MSCHAPv2 (Tiger)
- Cisco VPN Client for Mac Integration

INFRASTRUCTURE INTEGRATION

- Open LDAP integration
- Open Directory integration
- Databases (MySQL, Oracle)

NETWORK INTEGRATION

- RADIUS-enabled systems
  – *VPN, Firewall, NAS*
- Web Servers & On-Demand Computing
  – *Apache, IIS, Citrix*

CRYPTOCard Secure Password® TECHNOLOGY
www.cryptocard.com

SEASON SPECIAL

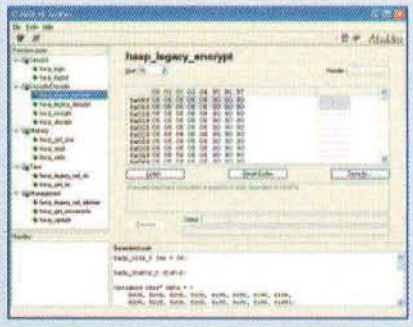
5-User Kit
* includes 5 tokens, CRYPTO-Server software & 30-day support
for only **$249**

Email **mac@cryptocard.com** and quote **SPMac249-01**

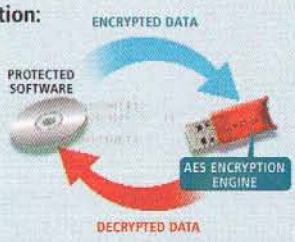
Offer available until January 31, 2006

3,248 hours typing code
184 hours finding that one bug
142 hours of meetings
108 pizzas
14 cancelled weekend trips
11 all-nighters
1 call protects it all!
HASP®
SOFTWARE DRM
#1 Choice of Software Developers

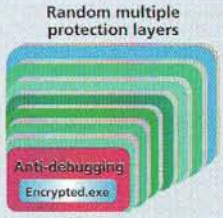
The new HASP family of products is the next generation in protection ensuring the highest level of security for your software. It provides an easy set of tools to automatically protect your software and implement new and innovative licensing options.



Strongest anti-piracy solution:
Our powerful, 128-bit AES encryption provides a strong locking mechanism that ensures you get paid for every copy of your software.



Robust envelope:
Automatically protects Windows executables, DLLs, .Net and Mac applications, securing the weakest link and ensuring your valuable IP remains unreachable. Multiple protection layers provide unequaled security.



Best overall solution as rated by an independent testing lab:
In KeyLabs tests, HASP HL outperformed the competition in security, ease-of-use, flexibility of tools and functionality, and compatibility across platforms and environments. A copy of the full KeyLabs report is available from our Web site at www.aladdin.com/hasp.

Category	Weight	Aladdin HASP HL	Sentinel UltraPro
Security	35%	85.1%	60.9%
Ease-of-use	25%	93.7%	77%
Licensing	20%	75%	75%
Flexibility of tools and functionality	10%	93.3%	71.2%
Compatibility across platforms and environments	10%	93.3%	71%
Overall Evaluation		**87.3%**	69.8%

Request a Software Developer Kit today at www.aladdin.com/hasp

Aladdin®
SECURING THE GLOBAL VILLAGE

North America: 1-800-562-2543, 847-818-3800 • UK • Germany • Israel • Benelux • France • Spain • Asia Pacific • Japan

©2005 Aladdin Knowledge Systems, Ltd. All rights reserved. Aladdin and HASP are registered trademarks of Aladdin Knowledge Systems, Ltd. Windows®, Mac OS®, Linux® are trademarks or registered trademarks of their respective holders.

Build Your Own Supercomputer From Your Macs Laying Around

How to employ Tiger's Xgrid to build clusters and to contribute to grid/cluster projects.

By Mary Norbury-Glaser

Grid vs. Cluster

Even in the Linux world, where cluster and grid computing are well established, there is often confusion about how to distinguish a grid from a cluster. Let's look at characteristics of each.

Grid computing uses the available resources of many individual, loosely coupled computers used by many different people across organizations. Grid computing makes use of idle computer time and unused disk space on different systems, often desktop computers (the SETI@home project, for example). Administrative tasks like system scheduling and job management are "distributed" since the computers are typically located over a wide area of multiple domains. There is usually no single system image; Windows, Linux and Mac systems often contribute to a single grid. Grid computing can be used for both high throughput computing (Apple/Genentech BLAST, for example) and high performance computing (the San Diego Supercomputer Center, the National Center for Supercomputing Applications, the Argonne National Laboratory, and the Max Planck Institute for Gravitational Physics collaborated in the largest grid computing demonstration of simulations involving Einstein's General Relativity equations).

Cluster computing, in contrast, involves a group of tightly coupled computers that work in parallel to share processing as if they were one machine with multiple CPUs. Administrative job management and a scheduling system are centralized in a cluster. Cluster machines also share a single system image. In other words, the collection of systems in a cluster appear as a single entity to the user, the DBA, etc. Cluster computing can be used for high performance computing (see description in grid computing above), load balancing and high availability computing. Load balancing describes the situation when processing activity is efficiently distributed among cluster members so no individual computer gets overwhelmed. High availability computing is implemented in situations that require maximum uptime and where availability of services is critical.

Think of grid computing as involving computers from myriad networks around the globe while cluster computing is a set of machines usually in one location involved in parallel computing with the goal to get them to look like a single virtual machine.

Xgrid Introduced

At MacWorld 2004 in San Francisco, Apple's Advanced Computation Group quietly announced the Technology Preview Release of Xgrid version 1.0 as a free beta download. Xgrid created quite a lot of excitement in ripples throughout the conference attendees who were involved in or interested in compute intensive applications. The software download included the Xgrid app and a basic local alignment search tool: Apple/Genentech BLAST (an open source DNA and protein sequence matching application) that enabled distributed searches on an Xgrid cluster. It touted the use of zero configuration (Rendezvous at the time, now Bonjour) to discover available resources on the network.

XGrid is based on a NeXT application called Zilla developed in the late 80's by Richard Crandall, a then NeXT employee and now a Distinguished Scientist at Apple. Zilla was the first community supercomputing application (screen saver type distributed computation system).

With the publicity of the Virginia Tech's Terascale Computing Facility and their System X built on 1100 dual processor, 2.0 GHz Power Mac G5 computers, the potential of low-cost supercomputing became an exciting possibility. Using commodity based or commercial off-the-shelf (COTS) hardware and free software, distributed computing became well within the reach of any organization with existing hardware.

Xgrid and Tiger

In April 2005, Apple introduced Mac OS X version 10.4 codenamed "Tiger" and included Xgrid in both the client and server versions of the operating system. Xgrid was streamlined and arrived in Tiger a different animal:

Xgrid in Tiger no longer limited job submission to the GUI; the client-side Cocoa API was made available to developers who were encouraged to assimilate Xgrid into their applications

instead of writing plugins. There are already several clients that have emerged from this addition to Xgrid: GridStuffer (a Cocoa client available at http://cmgm.stanford.edu/~cparnot/xgrid-stanford/html/goodies/GridStuffer-info.html) and PyXG (a Python interface to Xgrid, http://hammonds.scu.edu/~classes/pyxg.html).

A frustrating change for many users of the Panther technology release of Xgrid is the loss of the Xgrid **controller** on the Tiger client. Tiger client still includes the Xgrid **agent** but the GUI **controller** has been moved to Tiger server (see the next section for definitions of Xgrid terms). However, when we build our own cluster, we'll have some options. Patience, dear reader...

Definitions and Simple Explanations

Client: the Client submits jobs to the Controller. This can be any OS X 10.4 computer or OS X Server.

Controller: the Controller receives the jobs from the Client(s), splits the jobs into tasks and submits the tasks to the Agent(s). The Controller then receives the results back from the Agent(s) and delivers the information back to the Client(s). Apple has moved the Controller to OS X Server but has left the command line equivalent for managing the Controller and job submission on Client.

Agent: the Agent receives tasks from the Controller, runs the computations and sends the results back to the Controller. The Agent can be either a 10.3 or 10.4 computer. One task per CPU can be run.

Node: any single OS X (10.3 or 10.4) desktop computer or OS X 10.4 Server on the network.

There is only one Controller but there can be many Clients and many Agents.

The Controller, Client and Agent can be run on the same machine for purposing of testing. This is not, however, an optimal scenario for real world application.

Participate in a Grid...

Grid computing allows for an environment where idle CPU cycles and storage space of thousands of networked systems can work together on a particular processing-intensive problem. Current projects range from mathematics (prime number searches), science (climate prediction models), life sciences (cancer research), to cryptography (cracking data encryption schemes). By joining a grid project, your home or work computer can contribute to solving a global challenge.

The simplest use of Xgrid is to add your Mac to an existing distributed computing grid project. Deciding to participate in a grid is as simple as looking online for available Xgrid projects: http://distributedcomputing.info/projects.html lists active and upcoming projects under different categories with details on supported operating systems.

For example, Rosetta@home (http://boinc.bakerlab.org/rosetta/) is a project that is attempting to predict and design protein structures in an effort to help cure human diseases. To join this grid, go to their home page, create an account (you'll receive a return Account Key via email which you'll need to confirm at the web site and later insert into the application), make sure your computer meets the system requirements, and download the BOINC client software (Berkeley Open Infrastructure for Network Computing, http://boinc.berkeley.edu/) in either the GUI or command line flavor. The client software enables computers with different operating systems to "talk" to the project server.

Select the `Projects` tab and click the `Attach to new project` button. You'll be prompted for the project URL and Account Key that you received when you created your BOINC account.

Based on your chosen preferences (processor, disk, memory and network usage, etc.), the project will start and you can keep track of the progress through the tabs at the top of the BOINC app window.

Instructions for the command line version, along with command line options are available here: http://boinc.berkeley.edu/download.php.

...Or a Cluster

In this tutorial, we'll be looking at Xgrid@Stanford, a project run by Charles Parnot, a postdoctoral fellow in the Molecular and Cellular Physiology Department at Stanford University. His research involves 3D modeling of G protein-coupled receptors to study heart disease and heart rate control. The Stanford project began in March 2004 with 8 computers running at about 4 GHz. By September 2005, they had nearly 500 registered agents with an average of 200 machines online at any given time, the cluster running over 200 GHz. Xgrid@Stanford's home page (http://cmgm.stanford.edu/~cparnot/xgrid-stanford/index.html) describes the project and details FAQs, Goodies and Latest News about the project. Note that the Stanford group calls their project a "cluster" because they are using Xgrid exclusively and all the participating agents are Mac OS X computers. To join Xgrid@Stanford, we'll start by introducing ourselves to Xgrid (note the team has provided separate instructions for Panther and Tiger):

Set up Xgrid on Tiger:

1. Open `System Preferences`.
2. Open the `Sharing Pref Pane`.
3. Make sure your computer has a unique name!
4. Highlight `Xgrid` from the list under the `Services` tab and click the `Configure` button. Do not check the Xgrid checkbox!
5. Under Controller, select `Use a specific controller:` and type in:

b161-g4.stanford.edu

6. Under `Agent` accepts tasks: choose `Always`.
7. Under `Authentication method:` select `None` in the pop down list. You may get a security warning dialog box that you may ignore. Press `OK` on the Xgrid configuration sheet.

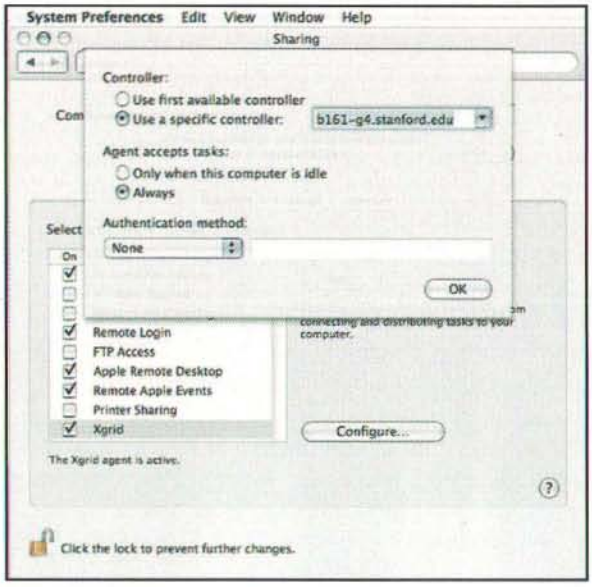


Figure 1: Configuring Xgrid Pref Pane

8. You will now be back at the `Sharing Pref` pane. Press the `Start` button.
9. Once the service is started, the `Start` button becomes a `Stop` button. If you need to change settings later, you will need to stop the agent, modify the settings and start it again.
10. Open the `Energy Saver` Pref pane and set `Put the computer to sleep when it is inactive for:` to `Never`.

Send an email to charles.parnot@gmail.com with your computer name, where you are from and where you heard about the project (reference MacTech, please!). The project team will let you know about system updates.

Download Dashboard Widget:

Download the Xgrid@Stanford Tachometer Widget from the Dashboard Widget section of Apple's website (http://www.apple.com/downloads/dashboard/status/xgridstanfordwidget.html) or from the developer's site (http://www.mekentosj.com/widgets/xgrid/).

Figure 2: Xgrid@Stanford Widget

The widget shows the total number of active or inactive agents or processors, the percentage of working agents and the current cluster speed.

Restore Screen Saver Functionality:

In the developer's preview of Xgrid, there was a nice tachometer screen saver feature that disappeared in Tiger. To get it back and view the Xgrid@Stanford widget as a screen saver, download either of these screen saver modules that allow Dashboard widgets to move about your screen during idle time: **Amnesty** (http://www.mesadynamics.com/amnesty_saver.htm) or **Dashsaver** (http://highearthorbit.com/software#DashSaver).

In this example, I've chosen to download **Amnesty**:

1. Drop the Amnesty Screen Saver.saver file into your `Screen Savers` directory (`~/Library/Screen Savers`).
2. Open the Desktop and Screen Saver Pref pane.
3. Select `Amnesty Screen Saver` from the list and click the `Options` button.
4. Select Xgrid@Stanford from the `Widgets` pop down list.
5. Set the time for when you want your screen saver to start.

Roll Your Own Cluster

You only need two computers to build a cluster with Xgrid: one agent/controller and one other agent. In this tutorial, we'll use three computers: one 12" Powerbook (named tiger12), one Mac mini (named mimimini), and one 15" Powerbook (our controller named norburym15).

JBL

A Harman International Company

iPod® OUTSPOKEN.

JBL ON STAGE™: SHARE MUSIC OUT LOUD WITH THIS HIGH-PERFORMANCE SOUND SYSTEM FOR YOUR MP3 PLAYER.

WWW.JBL.COM/JBLONSTAGE

OSA International, Inc. iPod is a registered trademark of Apple Computer, Inc.

Network the Macs Together:

If your Macs are not already networked together, you can easily do so using a 10/100Base-T Ethernet hub and a cat-5 Ethernet patch cable for each Mac. Connect each Mac to a hub port. Your Macs are also networked if you are sharing an Airport connection but you'll obviously suffer speed degradation in this scenario.

Verify that Bonjour is enabled (in `/Applications/Utilities/Directory Access.app`) on all Macs.

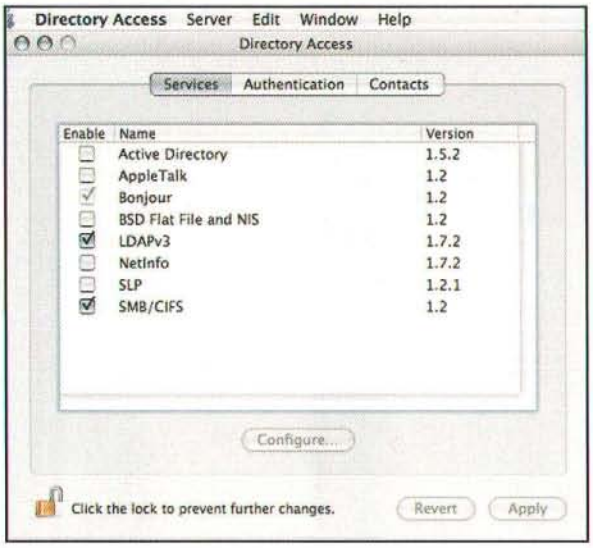


Figure 3: Bonjour in Directory Access

Configure the Agents (all Macs):

Open the `Sharing` Pref pane, highlight `Xgrid` from the list under the `Services` tab and click the `Configure` button. Select these options: `Use first available controller`, `Always`, `none` for authentication.

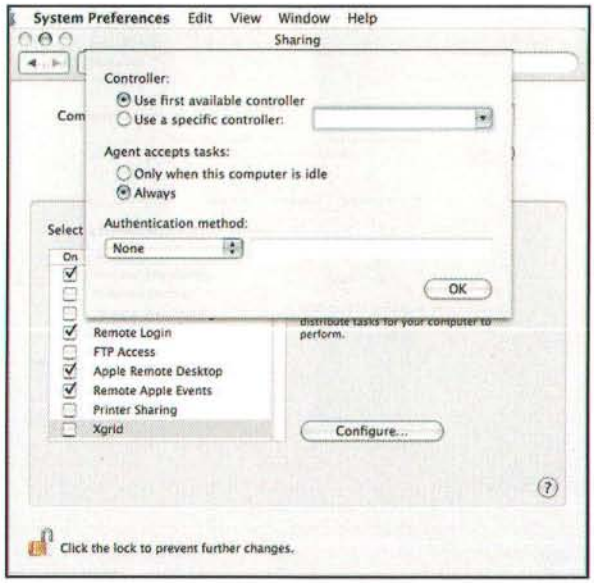


Figure 4: Configure the Agent

Click the `OK` button. You'll get a security warning; ignore it for now.

Click the `Start` button to turn on Xgrid Sharing.

Configure the Controller:

Remember I mentioned previously that the GUI controller has been moved to Tiger server? We have two options for restoring this functionality in Tiger client: we can download **XgridLite** (http://edbaskerville.com/software/xgridlite/), a $15 shareware add-on to Tiger Client's System Preferences or we can use the Terminal to control starting and stopping the controller.

XgridLite has some nice, basic features: it can turn the controller on and off and you can set passwords for client and agent authentication.

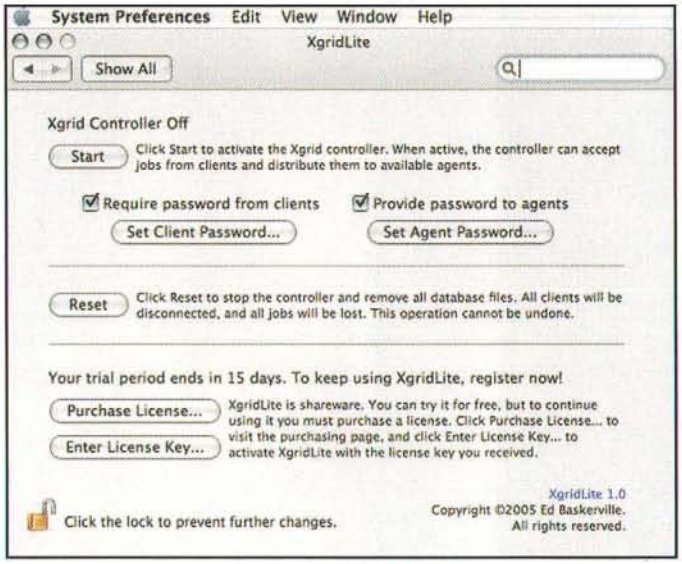


Figure 5: XgridLite

But since this is MacTech after all, let's do it from the Terminal using the `xgridctl` daemon. `Xgridctl` syntax comes in this flavor:

```
xgridctl status target on|off|start|stop|restart
```

where the target can be either `c`, which indicates the controller, or `a`, to indicate the agent. `On/off` refers to launching the daemon at startup and `start/stop/restart` explicitly controls the daemon.

Edit the file `/Library/Preferences/com.apple.xgrid.controller.plist` by changing the `Agent Authentication` string and the `ClientAuthentication` string from `Password` to `None`. Don't forget to use `sudo`!

```
sudo /usr/sbin/xgridctl c on
```

modifies the `launchd` configuration to run the controller daemon at startup and this command will start the controller:

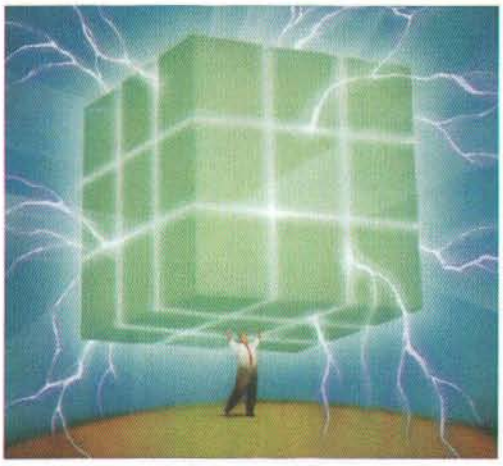
```
sudo  /usr/sbin/xgridctl c start
```

To use password authentication, you'll have to do a few more tweaks:

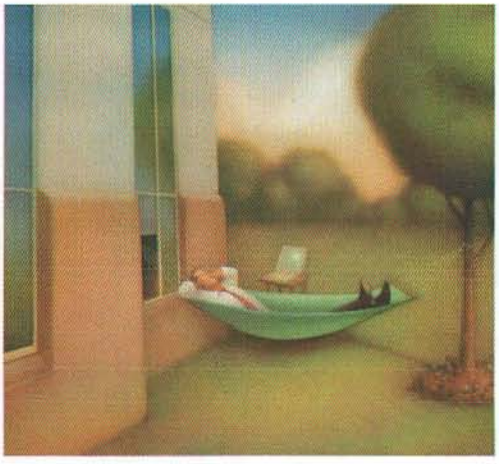
1. Edit the file `/Library/Preferences/com.apple.xgrid.controller.plist` by changing the `Agent`

Innovations by InterSystems

Rapid development with robust objects



Lightning speed with a multidimensional engine



Easy database administration

Massive scalability on minimal hardware

No More Object-Relational Mapping.

Caché is the first multidimensional database for transaction processing and real-time analytics. Its post-relational technology combines robust objects and robust SQL, thus eliminating object-relational mapping. It delivers massive scalability on minimal hardware, requires little administration, and incorporates a rapid application development environment.

These innovations mean faster time-to-market, lower cost of operations, and higher application performance. We back these claims with this money-back guarantee: *Buy Caché for new application development, and for up to one year you can return the license for a full refund if you are unhappy for any reason.** Caché is available for Unix, Linux, Windows, Mac OS X, and OpenVMS – and it's deployed on more than 100,000 systems ranging from two to over 50,000 users. We are InterSystems, a global software company with a track record of innovation for more than 25 years.

InterSystems CACHÉ™

Try an innovative database for free: Download a fully functional, non-expiring copy of Caché, or request it on CD, at **www.InterSystems.com/Cache14HH**

* Read about our money-back guarantee at the web page shown above.

© 2005 InterSystems Corporation. All rights reserved. InterSystems Caché is a registered trademark of InterSystems Corporation. 10-05 CacheInno14MaTe

Authentication string and the Client Authentication string to use Password.

2. Go to the Sharing Pref pane for Xgrid and select the Password option for Authentication method: and enter the password you wish to use.
3. Reset the agent via the GUI (stop/start). This creates the file: /etc/xgrid/agent/controller-password
4. Copy the file:

```
sudo cp /etc/xgrid/agent/controller-password
/etc/xgrid/controller/agent-password
sudo cp /etc/xgrid/agent/controller-password
/etc/xgrid/controller/client-password
```

5. Start and stop both the agent (via the GUI) and the controller (via xgridctl as above).

The Xgrid Admin.app allows GUI management and monitoring of the controller and it's part of the Server Admin Tools. Server Admin Tools 10.4 can be downloaded and installed on a Tiger client machine (http://www.apple.com/downloads/macosx/apple/serveradmintools104.html). Put this on your designated controller (mine is norburym15).

Launch Xgrid Admin.app and you will see a sheet asking to enter or choose a controller. You should see the name of the computer that you started the controller on using xgridctl (norburym15, in my case).

In The Xgrid Admin window Click the Connect button and you'll see the Overview:

In the Overview window Click the Connect button and you'll see your agents listed under the Agents tab:

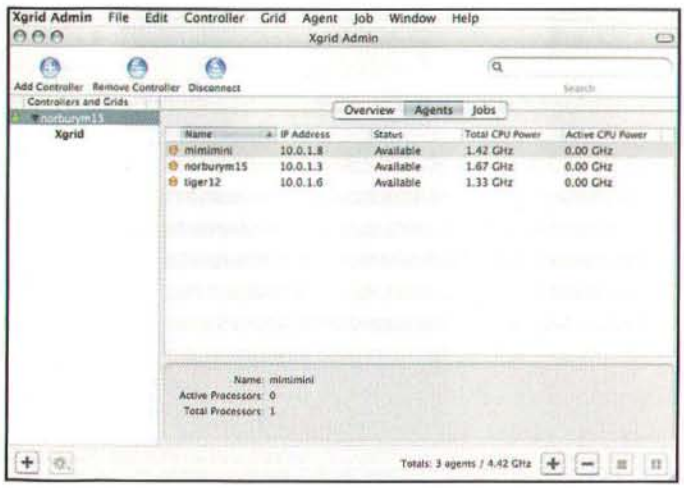


Figure 6: Xgrid Admin Agents Tab

There are no jobs listed under the Jobs tab yet so let's add one!

Add A Job Through the xgrid CLI Client:

Let's move over to my 12" Powerbook (as Client), open the Terminal and "talk" to the Controller (norburym15). First, we'll query to see what grids are available:

```
tiger12:~norburym$ xgrid -h norburym15.local -grid list
```

Our syntax includes the xgrid command, -h for hostname of the controller ("norburym15.local", since I'm using an Airport wireless network here at home), and the options -grid list to give us the list of available grids on our local network. The result is:

```
{gridList = (0); }
```

If we had two grids available, we'd see this result:

```
{gridList = (0,1); }
```

But we only have one grid and from the GUI above, we know it's called the default Xgrid. Let's take a look at information we can get from the CLI:

```
tiger12:~norburym$ xgrid -h norburym15.local -grid attributes
-gid 0
```

We see this for our result:

```
{gridAttributes = {gridMegahertz = 0; isDefault = YES; name =
Xgrid; }; }
```

The returned attributes include the current workload (gridMegahertz): 0 because there are no jobs running; whether the grid is the default one for the controller (isDefault): YES; and the name of the grid (name): Xgrid.

Let's run a job!

```
tiger12:~norburym$ xgrid -h norburym15.local -job run /bin/sh
-c cal 2005
```

This command initiates a job that calls the default month's calendar. When we hit enter and take then run over to take a quick look at the Xgrid Admin GUI app on our controller (norburym15), we can see some information while the job is running. Here's the Overview tab with a job running:

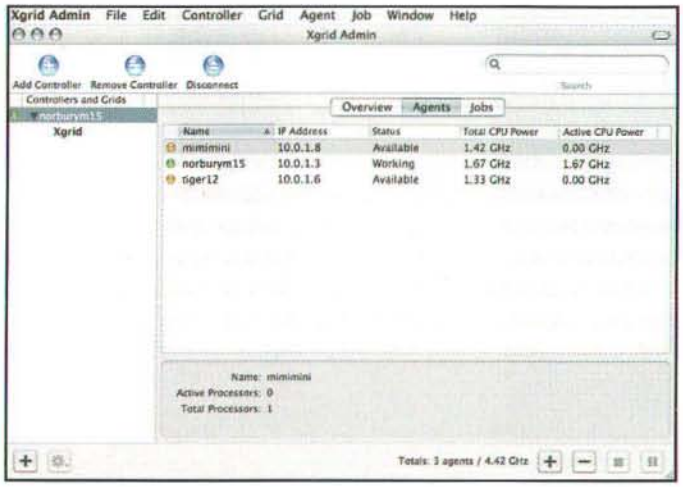


Figure 7: Xgrid Overview During Job Run

Click the Agents tab:

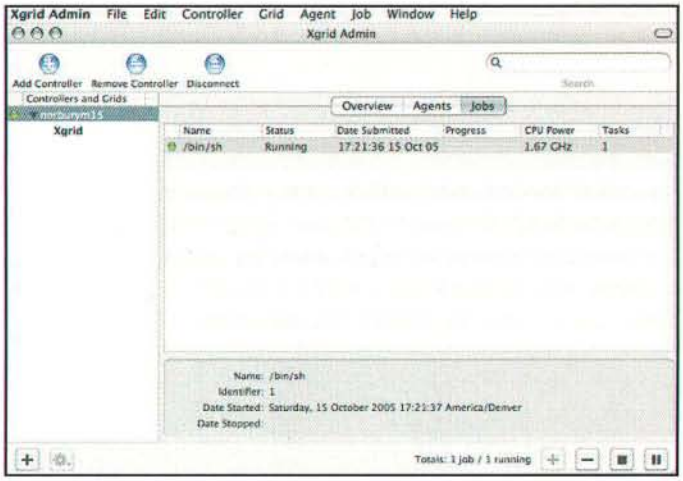


Figure 8: Xgrid Admin Agents During Job Run

And now click on the `Jobs` tab:

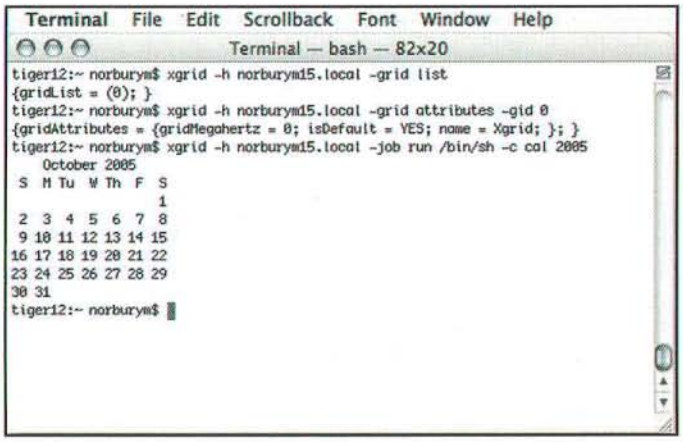


Figure 9: Xgrid Jobs Tab During Jobs Run

Finally, here is a screenshot of all the commands we've used to query the grid and initiate the job, along with the results from the job request:

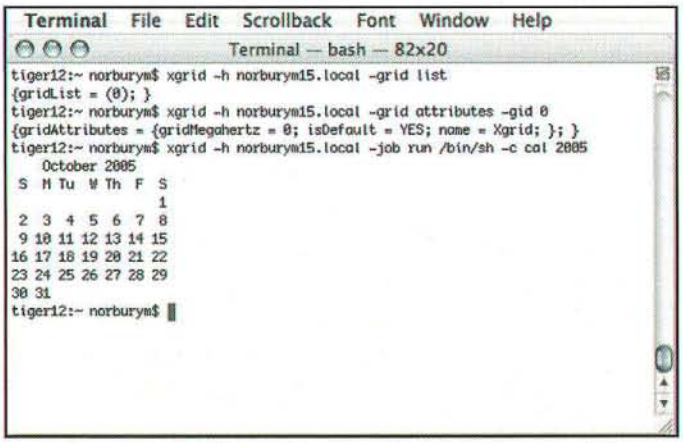


Figure 10. Xgrid CLI Client on tiger12 Client Machine

Take a look at the `xgrid` man pages for more options and some very good examples (rare for man pages but then Apple wrote this one!).

MACTECH® domains

Register

Get your .COM or any other domain name here!

FREE with every domain:

- FREE! Starter Web Page
- FREE! Getting Started Guide
- FREE! Complete Email
- FREE! Change of Registration
- FREE! Parked Page w/ Domain
- FREE! Domain Name Locking
- FREE! Status Alert
- FREE! Total DNS Control

Just visit
www.mactechdomains.com
to register for your domain today!

Starting at $3.99

Now...

Apple's Server Solutions site has a large section devoted to their proprietary tools for cluster computing (http://www.apple.com/xserve/cluster/). They provide ample information on their cluster technology solutions: G5, Xgrid, Xcode, Shark (performance optimization tool) and the Accelerate framework, as well as their cluster products: Apple Workgroup Cluster, Xserve G5 Cluster Node, Xserve Raid, OS X Server and Xsan. Their growing suite of integrated products provide robust and inexpensive solutions for customers who require solutions to their compute intensive application needs.

...And the Future

Steve Jobs' announcement in his WWDC 2005 keynote that Apple will deliver Macs with Intel microprocessors by Summer of 2006 ignited a considerable amount of discussion and speculation. The move from the PowerPC chip is controversial, no less for the apparent alliance with a company that has traditionally been aligned with "The Dark Side". This has, of course, sparked quite a bit of online discourse about running Windows on Intel-based Macs.

Apple is expected to use the Pentium M chip inside their next generation of desktop and portables because of its low power consumption and high performance. The Pentium M is geared to deliver more performance per clock cycle in order to consume less energy, which will ultimately make it run cooler than Pentium 4 chips. Of note, virtualization technology is built into Intel chips which will allow the machines to be partitioned to run different types of software like Windows or Linux at once, on top of Mac OS X. And hardware virtualization enables a system to run at near full-speed.

Apple has issued several statements saying they will not prevent Windows from running on Intel-based Macs but they will not allow the Mac OS to run on non-Apple machines. In fact, the Macs shipping as part of the Developers Transition Kit sport a security chip called the Trusted Platform Module that contains an encrypted serial number that verifies the OS is running on Apple hardware.

An interesting aspect of this is the potential development of cluster and grid virtualization. Traditional virtualization systems like VMWare emulate a PC down to the hardware devices. In other words, VMWare simultaneously creates multiple x86 virtual computers (guest virtual machines) and each virtual machine (VM) has it's own virtual CPU, disk, memory, etc. and all the virtual hardware is mapped to your computer's real hardware (the host machine). One limitation is significant emulation overhead and little to no opportunity for optimization. The VM (virtual machine) runs multiple kernels and full installations. Running virtualization on a cluster creates an environment with automatic load balancing via process migration: the master node provides system services while the compute nodes run the application. This scales up performance. Remember the difference between clusters and grids: with a cluster, there is a single point of administration while in a grid system, there are often different operating systems in different domains. While the cluster model provides a simpler, more well defined scenario, virtualization on grids will see advances in the future and Intel-based Macs could very well fill the growing need for low cost, commodity based distributed computing solutions.

MT

About The Author

Mary Norbury-Glaser is IT Director at the Barbara Davis Center for Childhood Diabetes, an affiliate center at the University of Colorado Health Sciences Center in Denver, Colorado. She has too-many-years-to-count experience in cross-platform systems implementation and administration in the education sector. You can reach her at norburym@mac.com.

MACTECH® Magazine
store.mactech.com/riskfree

Serve the MEDIA GENERATION

Equilibrium's DeBabelizer® and MediaRich® Servers: the automated media generation solutions.

Harnass the power of Mac® OS X and OS X Server to process your images, animations and DV for websites, mobile devices, and print with lightning speed. Serve up multiple digital media processing jobs and deliver the highest-quality results. Generate media derivatives from a self-service DAM portal. Magnify your online sales with dynamic image zoom and pan. When you need to automate your media workflow, Equilibrium's server solutions deliver.

Call 1.415.332.4343 to save time and money now. Visit www.equilibrium.com for free trials and stop by our booth #413 at Macworld SF 2006 to see the next generation in media automation.

equilibrium®

automated media processing solutions

© 2006 Automated Media Processing Solutions, Inc. DBA Equilibrium. All rights reserved. U.S. Pat. Numbers 6,792,575 and 6,964,009 for automated media processing and delivery. DeBabelizer and MediaRich are registered trademarks of Equilibrium. Mac and the Mac logo are trademarks of Apple Computer, Inc., registered in the U.S. and other countries.

Providing Progress Feedback During Script Execution

Many AppleScripts do not provide progress updates to the user during processing. Most of the time, when a script is run, it simply performs the appropriate tasks "behind the scenes," so to speak. If run as an application, a script may appear in the Dock when launched. However, this hardly provides detailed information to the user about what is actually occurring. Sometimes, a script may not need to provide progress updates to the user. However, there are situations when providing such feedback is a good idea.

In this month's column, we will walk through the process of creating a script that will provide visual progress information to the user during processing. The script we will create will save selected email messages in Mail as text files into a user-specified output folder. Since the script will have the ability to process multiple selected email messages, we will write our code to provide a visual indication to the user of which message is currently being processed. Once you learn how to provide this type of feedback, then you can begin integrating this same technique into your other scripts, making them more user friendly.

If you followed along with some of my past *AppleScript Essentials* columns, then you are probably familiar with AppleScript Studio, a feature set of Xcode and Interface Builder, the Mac OS X developer tools that come with OS X. AppleScript Studio provides a way for developers to build AppleScript-based applications, complete with robust interfaces. In this month's column, we will use AppleScript Studio to add a progress interface to our script, complete with text feedback and a progress bar.

Displaying a Basic Progress Interface with AppleScript

Before we get started with AppleScript Studio, let's discuss how to provide progress information to the user in a non-AppleScript Studio-based script.

In some cases, taking the time to construct an AppleScript Studio application may not be the best solution. For example, your script may be very simple, or it may be an existing script that is too complex to warrant conversion to AppleScript Studio at this time. You may just want a quick and easy way to provide feedback to the user. In these types of situations, the easiest method is to make use of the `display dialog` command, which can be found in the ***User Interaction*** suite of the Standard Additions scripting addition that is installed with Mac OS X.

Using the `display dialog` command, you can configure a script to display text messages in a no-frills dialog window at various times during script execution. The following example code demonstrates how a `display dialog` command can be used to provide such feedback.

```
set theOutputFolder to (choose folder with prompt "Select an
output folder:") as string
tell application "Mail"
  set theSelectedMessages to selection
  set theMessageCount to count theSelectedMessages
  repeat with a from 1 to theMessageCount
    display dialog "Processing message " & a & " of " &
theMessageCount giving up after 1 with icon note
    set theMessageContent to content of item a of
theSelectedMessages
    set theArchivePath to theOutputFolder & "Archived
Message " & a & ".txt" as string
    set theArchiveFile to open for access theArchivePath
with write permission
    set eof of theArchiveFile to 0
    write theMessageContent to theArchiveFile
    close access theArchiveFile
  end repeat
end tell
```

In the previous code, the first line of the script will prompt the user to select an output folder. Next, the script will retrieve a list of selected email messages in Mail. The script will then proceed to loop through each of these messages. During each repeat loop cycle, the script will retrieve the content of the current message, and save it into a file in the output folder chosen by the user.

As you can see, in order to provide visual feedback to the user during processing, I have made use of a `display dialog` command within the script's repeat loop. This will cause the script to display a message to the user each time the script begins processing one of the selected email messages. I have configured the `display dialog` command to indicate the current message count, as well as the total count of selected messages. See figure 1.

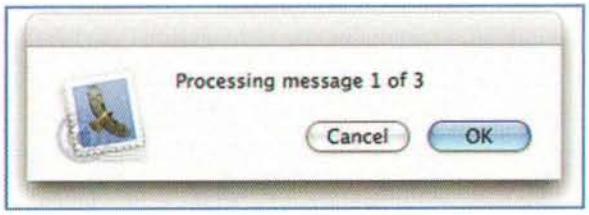


Figure 1. A Basic Script Progress Dialog

I have also configured the `display dialog` command to automatically dismiss the dialog after it has been displayed for 1 second. This will ensure that the script can proceed automatically with further execution, without the user being required to actually click a button in order to proceed.

Upon the successful execution of the previous code, the output folder specified by the user should contain text files containing the contents of the selected email messages. See figure 2.

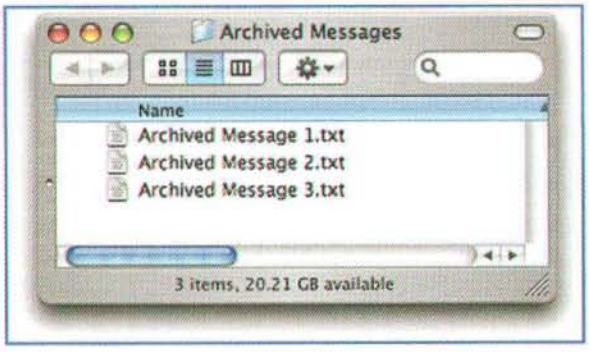


Figure 2. Archived Email Messages

The `display dialog` command can also be used independently, rather than within a repeat loop to indicate various tasks that are being performed by the script.

Now that we have discussed providing basic progress information to the user, let's move on to AppleScript Studio. In the remainder of this month's column, we will walk through the

Keep Your
Rack Mount
Equipment *Quiet*
And
Locate It
Where You *Need It*

XRackPro2

Noise Reducing
Rack Mount
Enclosure Cabinet

- Apple Approved and Recommended
- Reduces Noise from Multiple Chassises Up to 75%
- Heavy Duty Locking Casters Add Mobility and Stability

GizMac Accessories, LLC.
XRackPro.com
(800) 475-1677

process of creating a script that will perform the exact same function as the previous code, only with a more robust interface.

Building the AppleScript Studio Project

You may recall that the first step in building an AppleScript Studio project is to create a new project in Xcode. Begin by launching Xcode.

Please note that the AppleScript Studio project covered in this article was developed using Mac OS X 10.4.2 and Xcode 2.1. Please be aware that new software versions often result in changes in AppleScript terminology. Therefore, if you are using software versions other than those that I have specified, your required terminology may differ slightly from that which I am using in this article.

Create a new project by selecting ***New Project...*** from the ***File*** menu in Xcode. When prompted, select a project type of ***AppleScript Application*** from the list of available project templates, enter a project name of ***Archive Selected Messages***, and specify an output folder for the project. Xcode will duplicate the ***AppleScript Application*** project template into the specified output folder, and open it for you. See figure 3.

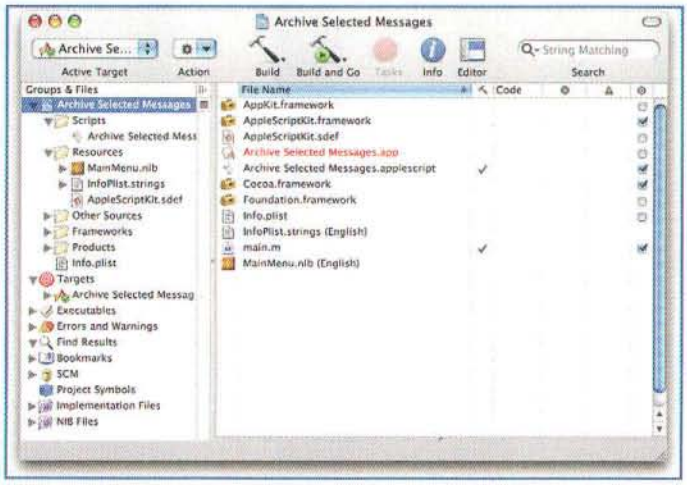


Figure 3. Archive Selected Messages Project Window

Building the Interface

Next, we will create an interface for providing progress information during processing. Double click on the ***MainMenu.nib*** file within your Xcode project window. This will open the project's default view in the Interface Builder application.

Designing the Window

By default, the project's interface should already contain an empty window view. This window will be the basis for our interface. Click on the window, and give the window a name by entering ***Archive Selected Messages Progress*** into the ***Window Title*** field in the ***Inspector*** palette. See figure 4. If the ***Inspector*** palette is not visible, select ***Show Inspector*** from the ***Tools*** menu in Interface Builder.

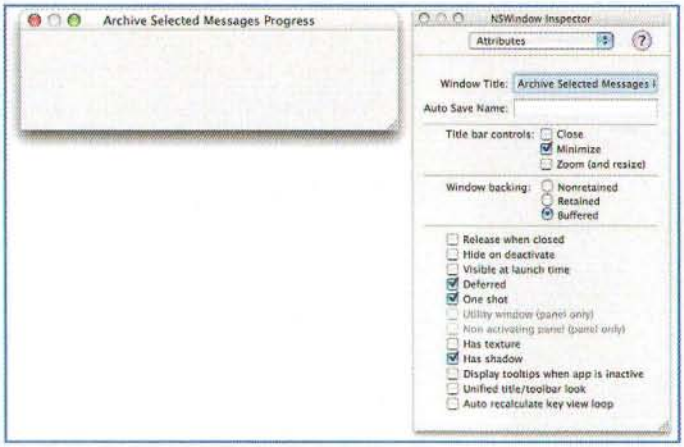


Figure 4. Preparing the Progress Window

Once you have specified the title for the window, de-select the ***Visible at launch time*** checkbox in the ***Inspector*** palette. This will cause the window to be hidden when the project is first launched.

You may want to make some other adjustments to the window configuration at this time, as well. For example, you may want to disable the ability for the user to close or zoom the window. Refer to figure 4 for the settings that I have specified for my progress window.

Next, click on the ***Cocoa Controls and Indicators*** tab in the toolbar of the ***Palettes*** window. If the ***Palettes*** window is not visible, select ***Palettes > Show Palettes*** from the ***Tools*** menu. Next, locate an ***NSProgressIndicator*** progress bar interface element in the ***Palettes*** window, and drag it into your project's window. See figure 5.

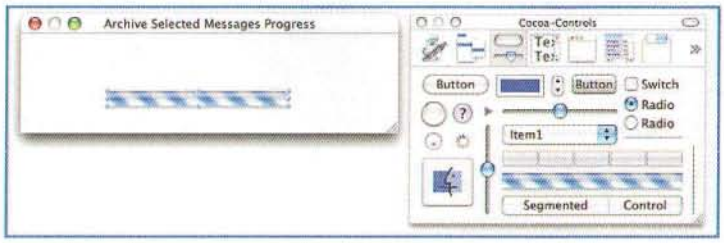
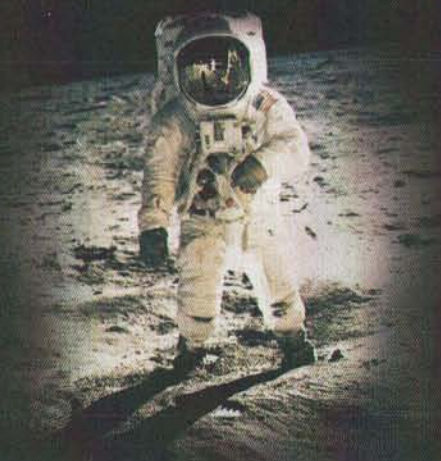


Figure 5. Adding an NSProgressIndicator

Next, click on the ***Cocoa Text Controls*** button in the toolbar of the ***Palette*** window, and drag an ***NSTextField*** into your interface window. Enter some default text, such as ***Waiting to process...*** into the text field's contents.

Continue to arrange and design the interface, making sure to adhere to Apple's standards for human user interface



One small step for man

Smart

One giant leap for data storage

NeoSmart

© 2005 Maxell Corporation of America

maxell

Expanding Memory & Mobility

Introducing NeoSMART (Super Maximum-capacity Advanced Reliable Tape) - Maxell's latest technological innovation.

With seven innovative advances, including an ultra-thin tape coating, nanometric metal particles, and superior servo-track writing, NeoSMART is expanding capacities from gigabytes to terabytes and beyond. Providing superior capacity, durability and reliability, NeoSMART is the ultimate solution for the ever-increasing demands of data storage. Just look for the NeoSMART symbol.

www.maxell-data.com.

Recordable Media Data Storage Portable Energy Technological Partnerships

guidelines, which can be found in the ADC Reference Library, both online and in Xcode's documentation. See figure 6 for an example of my completed interface design.

Figure 6. Example Progress Interface Design

Preparing the Interface for AppleScript Interaction

Once you have finished designing your progress interface, the elements that make up the interface must be prepared to interact with the AppleScript code within your project. To do this, you must assign AppleScript names to various interface elements, as well as configure certain elements of the interface to respond to event handlers.

First, we will assign an AppleScript name to the main window itself. To do this, click on the window to select it. Next, choose *AppleScript* from the popup button at the top of the *Inspector* palette, and enter the name *Progress Window* into the *Name* field. See figure 7. The process of assigning AppleScript names to the other interface elements will be the same.

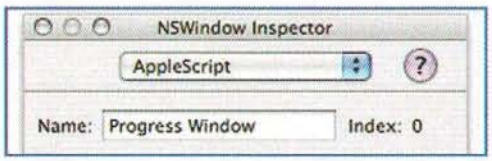


Figure 7. Assigning an AppleScript Name to the Progress Window

Select the progress indicator bar and text field in the window, and assign AppleScript names of *Progress Bar* and *Progress Text*, respectively.

As the last step in configuring the progress interface, we must configure the solution to run our AppleScript code when launched. This will be done by enabling an event handler. To do this, first select *File's Owner* in the *MainMenu.nib* window. A list of available event handlers will be visible on the *Inspector* palette, beneath the *Name* field. Enable the *launched* event handler by selecting its checkbox in the list of event handlers. Next, link the event handler to the AppleScript code in your project by selecting the checkbox next to *Archive Selected Messages.applescript* in the *Script* area at the bottom of the *Inspector* palette, beneath the event handler list. See figure 8.

MacResource
the.mac.resource

Wireless Products
Airport Cards
Extreme: $69.99
Standard: $159.99
Routers: $59.99

Accessories
3RD PARTY Wireless Mice $19 to $39
Apple Mice or Kbds $19 to $59
Kensington 4-btn Mice: $19

Parts for Every Mac System!

Logic Boards
G3 B&W Rev2: $99
G4 PCI : $129
G4 AGP Graphics: $219
G4 Gigabit: $349
G4 Digital Audio: $329
G4 Quicksilver: $499
G4 Mirrored Doors: $489
iMac Logics from $199
Beige G3 Series: $49

Power Supplies
Beige G3 Desktop: $49
Beige G3 Tower: $69
Blue & White Tower: $89
G4 Quicksilver: $179
G4 Digital Audio: $189
G4 Gigabit Ethernet: $199
G4 AGP Series 237w: $219
G4 AGP Series 208w: $179
G4 Mirrored Doors: $199

Logic board and power supplies require exchange.

Thousands of parts in stock for all Mac systems!

www.mac-resource.com

Mac Systems
G4/466Mhz: $249
G4/733Mhz: $699
G4/1000Mhz: $879
G4/1.25 OS9:$1099

New Systems Arriving Daily! Call for latest Stock.

eMac 700Mhz: $399
eMac 800Mhz: $449
eMac 1GHz: $629

iMac G4/700MHZ 15": $499
iMac G4/1.0Ghz 17": $749
iMac G4/1.25GHZ 20" $899

Xserves!!!
G4 1.0 ghz DP 512/60gb/CD NO SW $899
G5 2.0 ghz 512/80gb/CD $1999
G5 2.3 ghz DP 1024/80gb/Combo $2999
G5 2.3 ghz DP Nodes from $1999
Xserve Raid 1.2 TB 7 X 180gig $3399
Xserve Raid 2.5 TB 14 X 180gig $4499
Xserve Raid 2.8 TB 7 X 400gig $4999
Xserve Raid 5.6 TB 14 X 400gig $6999
Cache, Back-Up Batteries & Drive Upgrades available on all units
Free Ground Shipping when you mention MacTech Magazine

Refurb Displays

Apple 17-inch LCD: $349
Apple 15-inch Studio LCD: $149
Apple 22-inch Cinema: $999
Apple 17-inch Studio CRT (VGA OR ADC): $49

Products are open box, refurbished or demo call for more information

1-888-Mac-Resource

PHONEPIPE™ VOIP Service

Residential Plans

The PhonePipe 500

$14.99

500 Minutes - US and Canada

*Additional Minutes 3.5 cents

The PhonePipe 900

$19.99

900 Minutes-US, Canada, Austria, Belgium, Chile, China, Denmark, France, Germany, Hong Kong, Ireland, Italy, Malaysia, Netherlands, New Zealand, Norway, Singapore, South Korea, Spain, Sweden, Switzerland, Taiwan, UK

*Additional Minutes 3.5 cents

The PhonePipe Unlimited

$24.99

Unlimited Minutes
US and Canada

The PhonePipe Unlimited International

$34.99

Unlimited Minutes-US, Canada, Austria, Belgium, Chile, China, Denmark, France, Germany, Hong Kong, Ireland, Italy, Malaysia, Netherlands, New Zealand, Norway, Singapore, South Korea, Spain, Sweden, Switzerland, Taiwan, UK

*All Plans Include Caller ID, Voicemail, and Three-Way Calling

Business Plans

Base Line

Starting at
$20.00 a Month

Includes:

- VM
- CallerID
- 3-Way Calling
- Call Forwarding
- Call Waiting Plus

*Long Distance Starting at 2 Cents Per Minute

Enhanced Line

Starting at
$30.00 a Line

Includes:

- Basic Line Features
- Microsoft Outlook Integration
- Find Me-Follow Me
- Simultaneous Ring
- Personal Web Portal

*Long Distance Starting at 2 Cents Per Minute

Unlimited Line

$49.99 a Line

Includes:

- Enhanced Line Features
- Unlimited Long Distance to US and Canada

*Long Distance Starting at 2 Cents Per Minute

*All Business Lines Include PBX Features: Call Transfer, Music on Hold (Customizable), Call Hold, 4 or 5 Digit Dialing

Optional Features: Auto Attendant, Call Center, Reception Console

www.PhonePipe.com
1-877-300-3035 Ext 8200

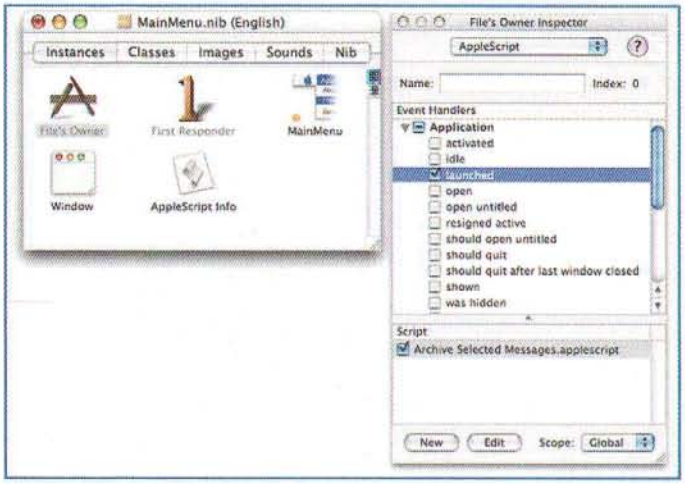


Figure 8. Enabling the Launched Event Handler

Adding the AppleScript code

Now that you have configured your interface, it is time to begin adding the AppleScript code into your project. Return to Xcode, and double click on ***Archive Selected Messages.applescript*** file in your project to begin editing the AppleScript code.

Preparing the Launched Event Handler

Since we configured our interface to respond to the `launched` event handler, we will need to add this handler to our project's code. If you saved your project's interface in Interface Builder, then this code may have been automatically inserted for you within your main project script in Xcode. If it does not exist, enter it as follows:

```
on launched theObject

end launched
```

Any code that is entered into this `launched` handler will be executed when our project application is launched. As we proceed, enter all code below within the `launched` handler.

Get the Output Folder

It is now time to begin adding the processing code to our project. Begin by entering the following code, which will prompt the user to select an output folder. Take note that this code is identical to that used in our non-AppleScript Studio example.

```
set theOutputFolder to (choose folder with prompt "Select an
output folder:") as string
```

Get the Selected Messages

Next, add the following code, which will retrieve a list of any selected email messages in Mail, and will then count the detected messages.

```
tell application "Mail"
  set theSelectedMessages to selection
end tell
set theMessageCount to count theSelectedMessages
```

Show the Progress Window

You may recall that we configured our progress window to not be visible on launch. The following code will now make this window visible to the user.

```
set visible of window "Progress Window" to true
```

Prepare the Progress Bar

We are now ready to begin preparing the progress bar within our window. First, we will set the `maximum value` property of the progress bar to the number of detected email messages.

```
set maximum value of progress indicator "Progress Bar" of
window "Progress Window" to theMessageCount
```

Next, in order to ensure that our progress bar will display incremental progress, we will set the `indeterminate` property of the progress bar to a value of `false`. A progress bar with an `indeterminate` property value of `true` will appear as a blue and white striped bar, as can be seen in figure 6, and this is not desirable for providing incremental progress.

```
set indeterminate of progress indicator "Progress Bar" of
window "Progress Window" to false
```

Looping Through the Selected Messages

We will now add code to loop through the selected email messages. Enter the following code into the script:

```
repeat with a from 1 to theMessageCount

end repeat
```

Updating the Progress Text

The following code should be added immediately within the repeat statement. It will update the text field in our interface to indicate the current message being processed. As in our non-AppleScript Studio example, this text will tell the user the current message count, as well as the total message count.

```
set contents of text field "Progress Text" of window
"Progress Window" to "Processing message " & a & " of " &
theMessageCount
```

Getting the Current Message's Contents

Next, add the code below, which will retrieve the content of the current email message.

```
tell application "Mail"
    set theMessageContent to content of item a of
theSelectedMessages
end tell
```

Stay In Control Wherever You Go.

Timbuktu® Pro

Whether you're at home or at work, using a Mac OS or Windows platform, Timbuktu Pro allows you to operate distant computers as if you were sitting in front of them. Transfer files and folders quickly and easily, and communicate by instant message, text chat, or voice intercom.

eCare

eCare's web-based remote access capabilities enable any user to securely place their virtual eyes and fingers directly on the remote client's desktop for collaboration, problem resolution, and improved customer satisfaction. Using only a web browser, eCare's secure and fast screen sharing, file transfer, live chat, URL push and live session recording make it the ideal solution for the conference room, classroom, or help desk – live, on demand.

Download an evaluation or buy it online: www.netopia.com
Call us at 1-800-485-5741

© 2005, Netopia, Inc. All rights reserved. Netopia, the Netopia design, and Timbuktu are registered trademarks belonging to Netopia, Inc., registered in the U.S. Patent and Trademark Office. All other trademarks are the property of their respective owners.

timbuktu®pro · eCare

netopia.

Saving the Contents to a File

Again, using the exact code from our non-AppleScript Studio example, add the following to your script. This code will write the content of the current message to a text file in the specified output folder.

```
set theArchivePath to theOutputFolder & "Archived Message " &
a & ".txt" as string
set theArchiveFile to open for access theArchivePath with
write permission
set eof of theArchiveFile to 0
write theMessageContent to theArchiveFile
close access theArchiveFile
```

Updating the Progress Bar

Finally, for the last section of code within the repeat loop, add the following code. This code will set the content of the progress bar to the current repeat loop increment, thus increasing the progress bar's display to accurately reflect the current number of messages processed.

```
set content of progress indicator "Progress Bar" of window
"Progress Window" to a
update window "Progress Window"
```

Please note that the last line in the preceding code will update the window, ensuring that the progress bar's interface is refreshed each time its content value is changed.

Completing the Code

To complete the handler, add a `quit` command at the end of the handler, just outside of the repeat statement. This will ensure that the application quits, once processing is complete.

```
quit
```

Next, wrap all of the code within the handler inside of a try statement, configured as follows:

```
try

on error theErrorMessage number theErrorNumber
  if theErrorNumber = -128 then quit
  error theErrorMessage number theErrorNumber
end try
```

This try statement will trap for a user cancelled error, error number -128, which would occur if the user clicks the *Cancel* button when prompted to select an output folder.

Now that your script is complete, the following example code shows how the completed `launched` handler should appear within your main project script.

```
on launched theObject
  try
    set theOutputFolder to (choose folder with prompt
"Select an output folder:") as string
    tell application "Mail"
      set theSelectedMessages to selection
    end tell
    set theMessageCount to count theSelectedMessages
    set visible of window "Progress Window" to true
    set maximum value of progress indicator "Progress Bar"
of window "Progress Window" to theMessageCount
    set indeterminate of progress indicator "Progress Bar"
of window "Progress Window" to false
    repeat with a from 1 to theMessageCount
      set contents of text field "Progress Text" of window
"Progress Window" to "Processing message " & a & " of " &
theMessageCount
      tell application "Mail"
        set theMessageContent to content of item a of
theSelectedMessages
      end tell
      set theArchivePath to theOutputFolder & "Archived
Message " & a & ".txt" as string
      set theArchiveFile to open for access theArchivePath
```


BMS

THE LAW OFFICE OF
BRADLEY M. SNIDERMAN

Need help safeguarding your software?

If you're developing software, you need your valuable work protected with trademark and copyright registration, as well as Non Disclosure Agreements.

Then, when you are ready to sell it, you can protect yourself further with a licensing agreement.

I am an attorney practicing in Intellectual Property, Business Formations, Corporate, Commercial and Contract law.

Please give me a call or an e-mail. Reasonable fees.

23679 Calabasas Rd. #558 • Calabasas, CA 91302
PHONE 818-706-0631 FAX 818-706-0651 EMAIL brad@sniderman.com

To: IT Department
From: Graphics Department
Date: Today 9:30 AM
Subject: HELP!

We need you to find a solution that is going to track and archive our production jobs and files from start to finish. And make them easy to find after we archive them. Even if our files are offline. And it needs to integrate with our **FileMaker Pro database**...

... by tomorrow.

Relax, We've Got You Covered...

Introducing

DigitalStorageManager™ 1.0

Intranet Search and Archive Management for Creative Workgroups

Digital Storage Manager is a high-capacity, high-performance file search and archive management system for creative, prepress and publishing workgroups. Deployable as either a stand-alone product or an integrated solution for FileMaker Pro, Digital Storage Manager provides real-time indexing of OS X and Windows file servers, automated metadata tagging, and powerful archiving capabilities.

Meta COMMUNICATIONS

Download a 21 Day Trial or Call to Schedule a Demo

Web: www.meta-comm.com Tel: 1-800-771-6382 Email: sales@meta-comm.com

© 2005 MetaCommunications, Digital Storage Manager is a trademark of MetaCommunications, Inc. All rights reserved. Additional company and product names may be trademarks or registered trademarks of the individual companies and are respectfully acknowledged. Software and licensing subject to change without notice.

```
with write permission
      set eof of theArchiveFile to 0
      write theMessageContent to theArchiveFile
      close access theArchiveFile
      set content of progress indicator "Progress Bar" of
window "Progress Window" to a
      update window "Progress Window"
    end repeat
    quit
  on error theErrorMessage number theErrorNumber
    if theErrorNumber = -128 then quit
    error theErrorMessage number theErrorNumber
  end try
end launched
```

Testing the Project

Now that our project is complete, it is ready for testing. To test the project, first launch Mail, and select multiple email messages. For best results, you may want to select a large number of messages, in order to ensure that the progress bar will increment properly. Next, select *Build and Run* from the *Build* menu in Xcode. If everything works as expected, your solution should launch, and you should be prompted to select an output folder. After choosing a folder, the project's interface should be displayed, and the selected messages should be processed and saved into the specified output folder.

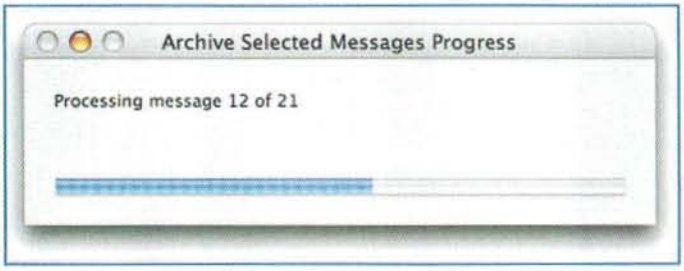


Figure 9. The Completed Progress Interface

Rumpus FTP

The Complete Internet File Transfer Server For Macintosh

Turn any Mac into a file transfer server...

- Simple installation & setup
- Dozens of advanced features
- Fast, efficient server
- Local or remote administration

To let anyone send or receive files...

- Transfer files with any Web browser
- Compatible with all major FTP clients
- Support clients on Mac or PC

Download A Free, Fully-Functional Trial Copy Today

http://www.maxum.com/

Other Options

In AppleScript Studio, a progress indicator may be displayed as a bar, as we have seen. However, a progress indicator may also be displayed as a spinner. See figure 10.

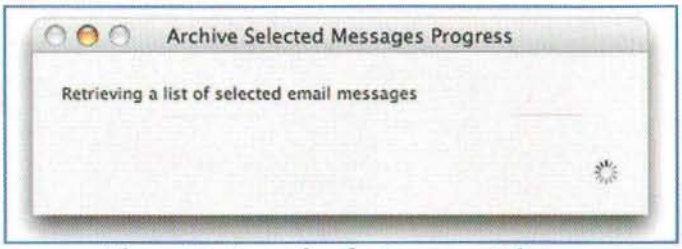


Figure 10. Example of a Progress Spinner

A progress spinner may be useful in situations where your code does not warrant providing incremental feedback to the user, yet you still wish to indicate that processing is occurring.

In Closing

The project discussed in this month's column should give you some basic ways to implement progress feedback in your scripts. Now, it is up to you to begin making your scripts more user friendly by providing such feedback to your users.

If you have difficulty getting any of the specified code to work, or if you prefer to review the actual project files, you may wish to download the example code. I have made the sample project discussed in this article available for download from my web site at the following URL:

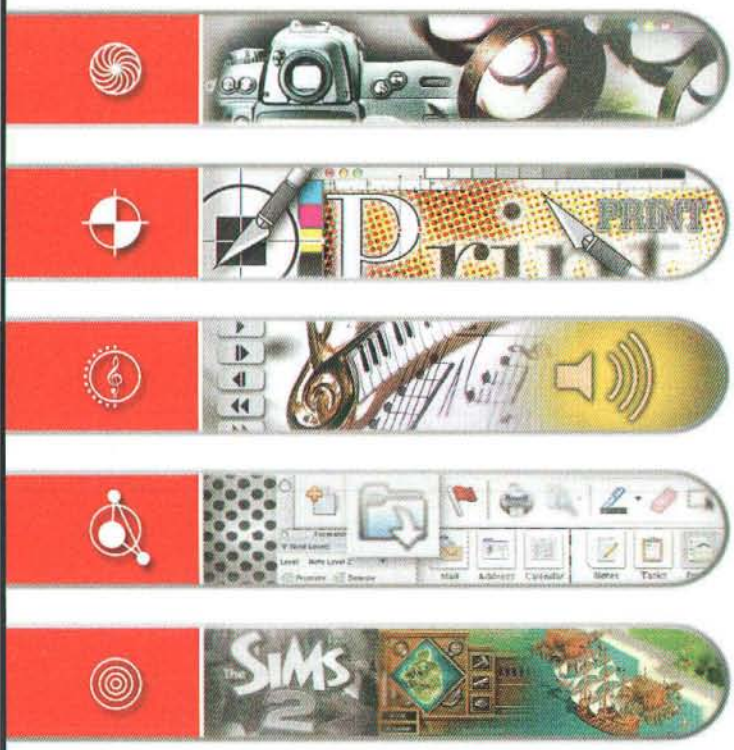
http://www.automatedworkflows.com/files/demos/MacTECH.01.06.Example.zip

Until next time, keep scripting!

About The Author

Ben Waldie is the author of the best selling books "AppleScripting the Finder" and the "Mac OS X Technology Guide to Automator", available from http://www.spiderworks.com. Ben is also president of Automated Workflows, LLC, a company specializing in AppleScript and workflow automation consulting. For years, Ben has developed professional AppleScript-based solutions for businesses including Adobe, Apple, NASA, PC World, and TV Guide. For more information about Ben, please visit http://www.automatedworkflows.com, or email Ben at applescriptguru@mac.com.

Macworld
Conference & Expo®



Macworld Conference & Expo is the #1 event for the Mac Community. Attendees can find the latest, newly announced products and learn to run their Macs more efficiently no matter what their needs are.

Macworld Conference & Expo San Francisco 2006 has your next Mac solution with training to boost your Mac abilities on OS X Tiger, music and audio, video, graphic design, photography, digital imaging, gaming, and more. Recognized industry experts provide professional and comprehensive information, networking opportunities abound and you'll get to meet hundreds of companies supplying the Mac products, innovations and applications that you want to see.

CONFERENCES: JANUARY 9-13, 2006
EXPO: JANUARY 10-13, 2006

THE MOSCONE CENTER
SAN FRANCISCO

M
SF

Flagship Sponsors:
Macworld Macworld.com

IDG WORLD EXPO

© 2005 IDG World Expo Corp. All rights reserved. Macworld, Macworld.com and Macworld Conference & Expo are trademarks or registered trademarks of International Data Group Inc. All other trademarks contained herein are the property of the respective owners.

macworldexpo.com
Register online using Priority Code: D2Ø01
by December 9, 2005 to take advantage of Early Bird Pricing!

Make eBay *Work For You* Instead Of You Working For It

MacTech's guide to eBay listing tools on the Mac

By Neil Ticktin, Publisher/Editor-in-Chief

eBay ... The Myth, The Pain, The Solution

We've all heard the stories, how someone made a bunch of money selling on eBay, or the eBay version of the stereotypical "fish story" that starts "I can't believe how much this guy paid for ...". eBay is part of our ecosystem today, like it or not. You can find some great stuff on it, and it's the tech way of going about doing a "garage sale" ... and you ***can*** make a bunch of money selling-off the stuff that you didn't think anyone would want.

The problem is that eBay is a pain. It's a lot of work. It takes time to get a good set of pictures, pull together the best descriptions, answer all the shipping information and other logistical questions up front, choose categories, choose pricing, the list goes on and on. Add to it that eBay is not a great customer support organization ... if you want to call them, you can't. If you want to ask a question, plan on doing it several times, as they try to answer with canned answers, which often are not the answer to your question.

This is where 3rd party tools come in. They are necessary if you are going to do anything beyond a listing or two on eBay. They will help prevent you from doing the wrong thing, and can help you understand what you are doing before you commit. And, for those who have a higher volume of items to sell, they give you a structure to work within and reduce redundant work.

Types of eBay Sellers

There are a few types of eBay sellers, and before you choose a third party listing tool, you need to identify the type you are ... or you will almost definitely choose the wrong solution. Be honest with yourself. It's easy to get caught up in wanting the most capable product, only to realize that you are selling 3 items and you are about to kill a fly with a bazooka.

Power Seller: If you're a power seller, you're the type of person that is working on eBay sales with some level of frequency ... from everyday to every couple of weeks. You may be selling lots of different items, or the same items over and over. In short, you're the kind of person that needs a process in place to keep from confusing things.

Burst Seller: If you periodically gather up all the things you have around the office or around the house, and do a "spring cleaning," you may have a number of items every few months that you want to sell. In this case, you need a piece of software that will help you build listings quickly, but you don't need as much of a process.

Experimenter: If you are just experimenting with eBay, have only a handful items to sell, or are more the type to see what kind of "junk" you find around the house you can turn into gold on eBay, then you are in our "experimenter" category. You might want to look at one of the really low cost solutions, or no solution at all (i.e., you may want to use the eBay interface to do all of your listing).

MarketBlast

MarketBlast is a high-end product published by 4D ... the database people. In short, if you are a Power Seller, then take a look at MarketBlast. If not, then this is not the product for you.

The good news about MarketBlast is that it's designed to give you a process. It's meant for the kind of person who needs to create inventories, profiles, and track not only a lot of sales and interactions, but also monitor how your marketing efforts are working. When you invest the time into the profiles, you'll find them to be flexible, capable, and allow you to control many different aspects of a sale.

Furthermore, as you would expect from a company that makes databases, you have the underlying power of a fully relational database. That means communications and payments can be associated with listed items, and listed items with inventory, and buyer histories across items. For the Power Seller, this is critical to keeping your sales efforts efficient in nature.

We started with an earlier version of MarketBlast, v1.11b. We had some trouble in that we corrupted the database on this version, but 4D did a good job of tech support, and repairing the database quickly. This product will get better over time. Things like having unusual file name characters (quotes and such) caused it to fire off erroneous error messages. But, these are bugs that we expect 4D will fix. Since then v1.12 was released and fixed some problems, as well as made refinements (like associating a picture with the inventory instead of the listing).

The one thing that you need to realize about MarketBlast is that it has a learning curve to it. You need to learn the process that it expects, and that you can use to make your selling efforts more efficient. You also need to learn what some of the interface items mean. For example, at one point I wanted to change the title of an active listed item. I made the edit, and then pressed the "Publish" button.

That had the unfortunate result of ending the active sale, and awarded it to the highest bidder of the time ... even though it was a week before the sale was complete, and the bid was nowhere near its value.

MarketBlast will benefit from continued refinement (like moving product weights to inventory items instead of needing to a profile, speeding entry of inventory items, etc...), but if you are a serious seller, moving a volume of products, and can put in the learning curve time, then it's your choice on the Mac (or Windows).

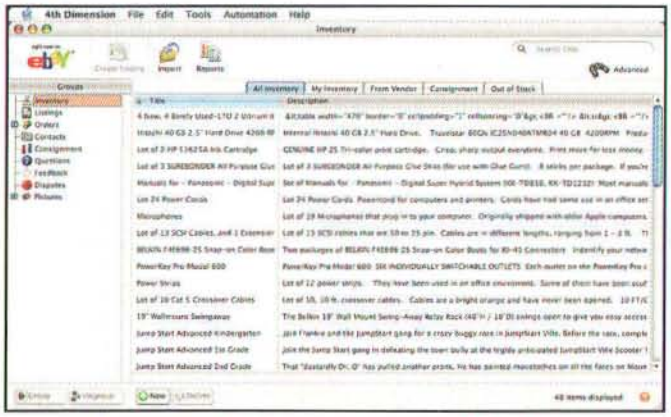


Figure 1. MarketBlast Inventory Window

It would be impossible to tell you all that MarketBlast can do in an article like this, but to give you a glimpse. It's a fully integrated solution that allows you to do multiple actions, in bulk, across eBay, eBay Motors, and eBay Stores. It helps you track inventory, and auto-updates quantities, and will alert you when the quantities reach certain levels.

You can manage vendor details, multiple sources per item, and group inventory in a variety of different ways. There are global profiles, and the sub-profiles ... and these are at the heart of customizing how you list products.

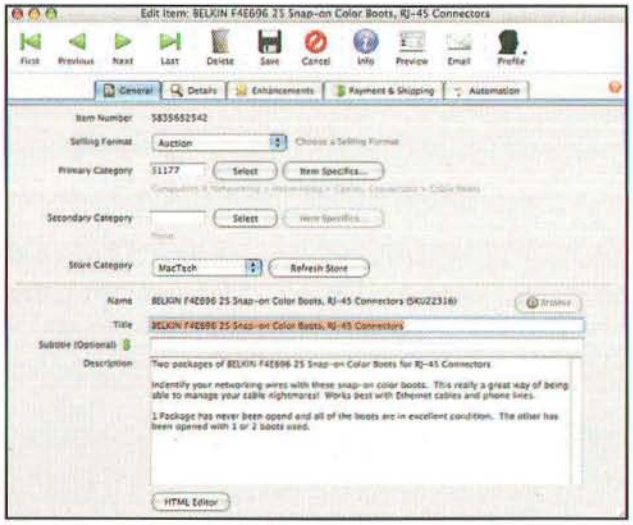


Figure 2. MarketBlast Edit Item Window

If you move between computers (say a laptop and a desktop), MarketBlast can be copied easily from one workstation to another. You can handle consignment, mailing labels, write scripts, automate emails, and send out alerts to consignors all automatically. There is an HTML editor, although their displaying of the page in a pseudo-browser is limited and somewhat flaky.

You can preview your eBay fees, and they do a solid job of supporting both ftp and Flickr (free limited, or low-cost unlimited storage of pictures). You can even watermark your images. There are tools for automatic relists, selling strategies, list reports, recurring listings, and auto price/quantity adjustments.

There's a full CRM (Customer Relationship Management) set of tools. For example, you can group your customers, do emails and newsletters, and more. And, with any volume business, there's going to be issues with deadbeat customers ... MarketBlast highlights them for you. Finally, you can do trending reports, show built-in ledgers, and analyze cash flow.

Again, if you ***are*** a Power Seller, take a good look at this product, but be prepared for a time investment to learn its ins and outs ... so that you can make the most of what it can do for you. If you are not a Power Seller, this is ***not*** the product for you.

For more information, see www.marketblast.com, Price: Regular: $149.99 Discount: $99.99, System Requirements: High-speed Internet connection, Macintosh, G3 or better processor, Mac OS X 10.3 or higher, 256mb RAM; Windows: Pentium II or better processor, Windows 2000, XP, 256mb RAM

iSale

The folks at equinox position iSale as the eBay component of Apple's iLife suite of applications. In a lot of ways, they are right. If you are a Burst Seller, or even an Experimenter, iSale is a great solution for you. As a Power Seller, you could use iSale, and it will do fine at what it does, but it's nowhere near the complete business solution that MarketBlast is.

First, I should tell you that I really like this product. It worked as I expected it to, and the nits that I have are only here to help you understand how minor my criticism is, and to make it a better product.

iSale's strength is that it really does behave as you would expect it would. It integrates well with iPhoto, and you can even use an iSight to take product shots. For Tiger users, there's a dashboard widget to watch your auctions (although it's relatively slow, and you can't grow the window to watch a larger list more easily). You can even use it with Automator, and create RSS feeds for your auctions.

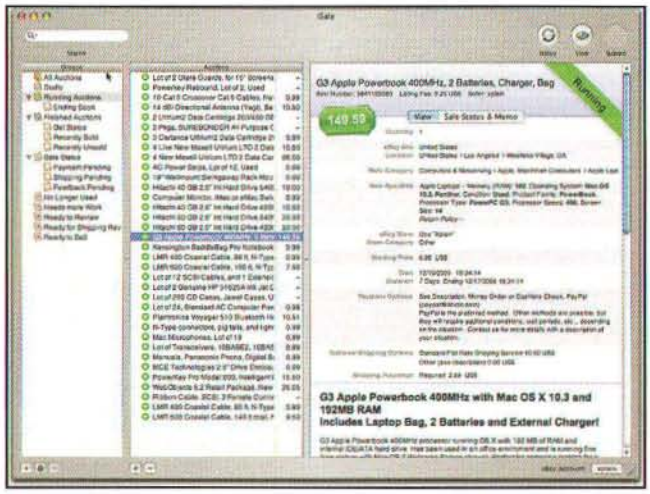


Figure 3. iSale Main Window

Categories are very easy to assign, you can see the listing fees, and you can even schedule things in iCal so you can manage your auction ending dates. They do a great job of using your .Mac account, or an ftp server. Auction status seems a bit slow at times, but that could easily be eBay as well. And, while it's not really a process work-horse, you can keep track of the after sales items like is something paid, sent and feedback status.

In the text editing, the application does a pretty good job of mixing both HTML and regular text, and still have it display well ... this makes life really nice for much quicker description creation.

iSale has several easy to use preferences that apply to new items created after you set the preferences. So, it's worthwhile to think them through before you start creating a lot of items. There's a number of settings, and the signature settings allow you to take care of anything that doesn't fit elsewhere.

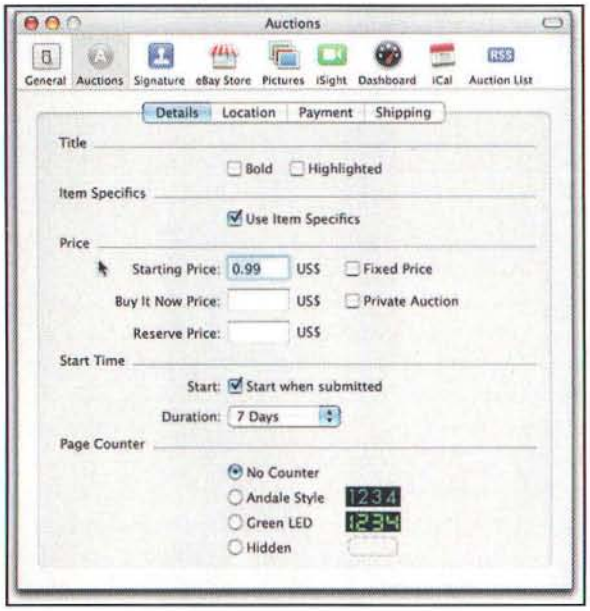


Figure 4. iSale Preferences

My nits

By far, the most irritating thing about this product is its registration. It's not a good experience and you have to jump through a lot of hoops just to get the software installed. Once it is installed, it's fine, but boy, it's almost bad enough to make you not want to do it. Contrary to what you might expect, it is ***not*** representative of the rest of the experience and you should drive on through it.

As part of equinox's desire to limit software piracy, they've made it difficult to move the software from one machine to another. For example, if you want to do a bunch of work on your desktop, and then move everything to your laptop to monitor the sales ... you may want to just buy a second copy. iSale has to be in the user's Application Support folder. It cannot be in the System's App Support folder, and the preferences for the app are not stored in the App Support folder, so if you go between machines, be careful.

I found that I really could use a flag to help sort based on certain status, and an internal number for identifying like units would also be useful. Both of these would be unpublished information, and simply for use by the user (not on eBay). For example, I was creating my own "workflow" and had to resort to using their Smart Lists based on the name of the item.

Dragging up and down in an editable text field is something that I'm use to doing on the Mac, but iSale forces me to use the scroll bars instead. I would like to see some sort of template that helped with descriptions. And, I would also like to see templates, or profiles, be applicable to items ***after*** they are created. Right now, it seems that many preferences only apply when you are first creating an item, and once it's created, changes in preferences would need to be manually applied.

I would like to be able to add picture slots prior to the last one being used ... so that if I know that I have 3 pictures, I can add two slots first, and then drop and drag them in without having to go back and forth to the item.

For more information, see www.equinux.com, Price: Single License - $39.95, "Double Pack – for two machines - $59.95. System Requirements: Macintosh Only, Mac OS X 10.3 or higher, Active Internet connection.

GarageSale

I had the least amount of "real life" time with GarageSale, but that shouldn't discourage you from this product. Like iSale, it has a very intuitive user interface, although unlike iSale which has a lot in one main window, GarageSale breaks things up a bit more and gives you ways to see things at a high level.

Figure 5. GarageSale Overview

And, GarageSale allows you to host pictures on both ftp or .Mac. You can do specialized listing designs, work offline, import directly from your camera, or work with iPhoto and iSight. There's a good integrated auction preview, and you can share templates with other GarageSale users if you'd like.

The heart of any eBay listing tool is its item editing. Here, GarageSale shines again ... a picture is worth a thousand words...

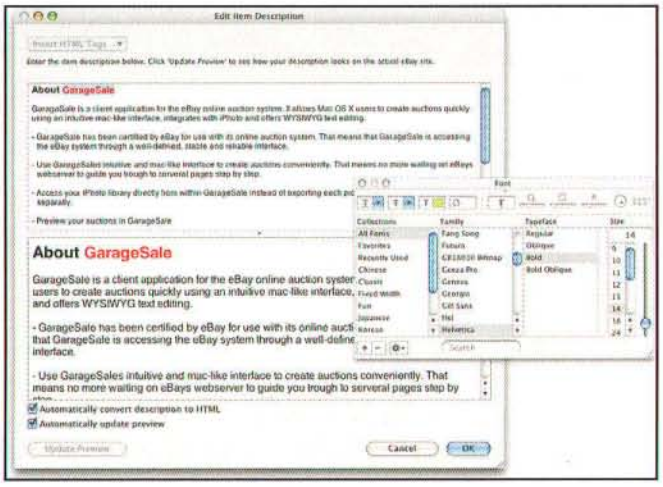


Figure 6. GarageSale Item Editing Window

For Tiger users, there's Automator, Dashboard, Core Image and Spotlight support as you would expect. And, you can use GarageSale for multiple eBay accounts, which will be important to some people.

For more information, see www.iwascoding.com/GarageSale/, Price: Single License: $24.99 Family – Up To 5 : $44.99. System Requirements: Macintosh only, Mac OS X 10.3.9 or better, Internet Connection.

CALC

CALC is an auction seller's productivity tool that's been around for a while now. The company's site claims it was the first Mac auction software for OS X. As an auction utility it combines listing creation with auction tracking in one neat interface. A notable feature is it allows you to create functional HTML auction listings without knowing HTML These ease of use makes this a good tool for beginners but it also has features that will appeal to more experienced users – you can add your descriptions, choose colors, select pictures and backgrounds, insert tables, style your text and preview your listing in your web browser before you list it. You can save and reopen listings for frequently listed items, which is great for items that are listed more than once.

For more information, see www.cycline3.com/calc/, Price: DVD Version - $19.99, CD Version - $14.99, Download Reg. - $9.99. System Requirements: Macintosh: Mac OS 8.5 - 9.2 or OS 10.2 or newer (Carbon application and PPC versions in .sit file); Windows: Pentium 150 or better and works with Windows 98 or newer.

eSeller

If you are just interesting in some of the back end duties of selling on eBay, there's eSeller - a shareware FileMaker template for keeping track of sales and generating packing slips for shipping your packages. According to its author Robert Hofemann. Version 3.3 and 3.4 combined have had almost a thousand downloads from Version Tracker. Bob asks only for a $5 shareware fee, so, if you're going to use it, please don't forget to register your software! Listed on Version Tracker are references to the developer's site and a short paragraph on the features of the actual template. There is a Mac and Windows version on the Developer's site along with a tutorial on how to use it. You can view the template by going to Version Tracker and looking up eSeller 3.4 at:

http://www.versiontracker.com/dyn/moreinfo/macosx/25787

Which for you?

We started off the article talking about the different types of eBay sellers there are. You really do need to start there, and then your decisions are relatively easy as to what tools you should use to make money on eBay.

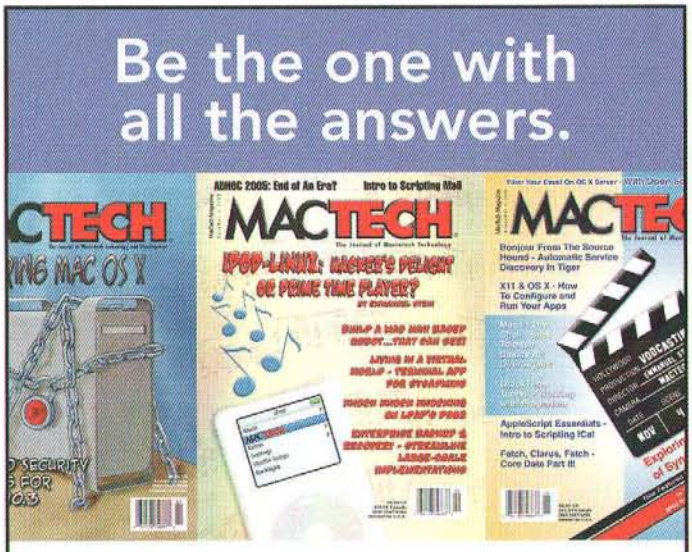
If you are a Power Seller, definitely take a look at MarketBlast. If you are someone that wants a solid tool, but not a full process management tool, then take a good look at iSale and GarageSale and see which fits you best. And, if you want a tool that will give you some help, but at a dirt cheap cost, check out CALC and eSeller.

All of these tools will give you a *much* better experience than trying to list directly on eBay using the web interface.

MT

About The Author

Neil is the Publisher/Editor-in-Chief of MacTech Magazine. Neil has been in the Mac industry since 1985. When Neil writes a review, he likes to put solutions into a real-life scenario and then write about that experience from the user point of view. That said, Neil has a reputation around the office for pushing software to its limits and crashing software/finding bugs.



MacTech is already read every month by tens of thousands of readers. MacTech readers represent the very heart and soul of the the Mac community. Join the crowd and sign up for MacTech today!

store.mactech.com/riskfree

UNIX UNDERPINNINGS ON THE MAC

WHAT TO USE AND WHAT TO AVOID

"Think Different" is a great campaign, however, it's nice to know - and downright important - that we share some things with others. Coincidentally, I'm sometimes reminded that I need to go back and cover the basics: the bare essentials. What better time to do that than the new year? (even though time is an abstract, it's a useful tool). So, while us Mac users like to run different, think different, be different, we can learn a lot by looking at what we have in common with other Unix systems.

Introduction

Although I don't hear it too much any more, when OS X first shipped, there was a, "OS X is **not** Unix" current from certain corners of the Web. While OS X certainly does some things differently than a traditional Unix, and continues to do so increasingly (especially under Tiger, which is all I refer to here), that doesn't make it "not Unix." As Apple "Applefies" more of the Unix that does underpin OS X, let's see what keeps us tied to our Unix brethren.

FreeBSD

The Unix conventions and utilities found in OS X are largely based on, and often taken straight from, FreeBSD. In Beta-vs-VHS thinking, perhaps the *BSDs are Beta to Linux's VHS: a loser in mindshare, but certainly technically superior. If you've used Linux, but not any of the BSDs, as a techie, you owe it to yourself to dive in and check it out. If you have a spare PC, or even an unused Mac, sitting around go download or purchase FreeBSD and give it a whirl. It'll also improve your OS X skills. Next time you're putting together a public facing web site on a budget, and are ready to reach for Linux, try FreeBSD instead.

cron

cron is the sub-system that is most likely completely ingrained in a sys-admin's skull. It would almost be impossible to be an effective admin without it. For the uninitiated, cron is a ***daemon*** that schedules jobs to run at certain times. You may have heard that cron no longer exists or runs as of Tiger. Nothing could be further from the truth. Apple's new subsystem, launchd, tries to take over all responsibilities for cron...and doesn't really succeed. A clean install of Tiger has no cron jobs installed by default. If you upgraded from Panther, you'll have your cron jobs carried over.

launchd is Apple's new system for managing daemons and 'agents' on a system-wide ***and*** per-user basis. It's a one-system-to-rule-them-all kind of thing that currently only half succeeds. I covered launchd in depth in the May 2005

Mac In The Shell column, pointing out both the advantages and disadvantages. If you're just joining us, I'll give you the top three reasons why launchd can't currently replace cron:

1. cron is way too engrained to go away anytime soon. This leads to...
2. launchd's reliance on plist files for configuration. As nice as the XML based plist is, there are enough tools out there that can easily update a crontab file thanks to it's plain-text based format.
3. Flexibility. While launchd ***can*** schedule a job to run at a certain time and date, it can't currently match cron's scheduling flexibility. As in "0 8-20/3 * * 1-5 root /path/to/program" which runs the given job every three hours, Monday through Friday only between 8am and 8pm.

In short, cron kicks butt! You certainly need to learn about launchd, but don't be afraid to schedule things with cron. It's not dead yet! If you're looking to learn a bit more about it, cron on OS X was covered in depth in the March 2005 MacTech.

xinetd

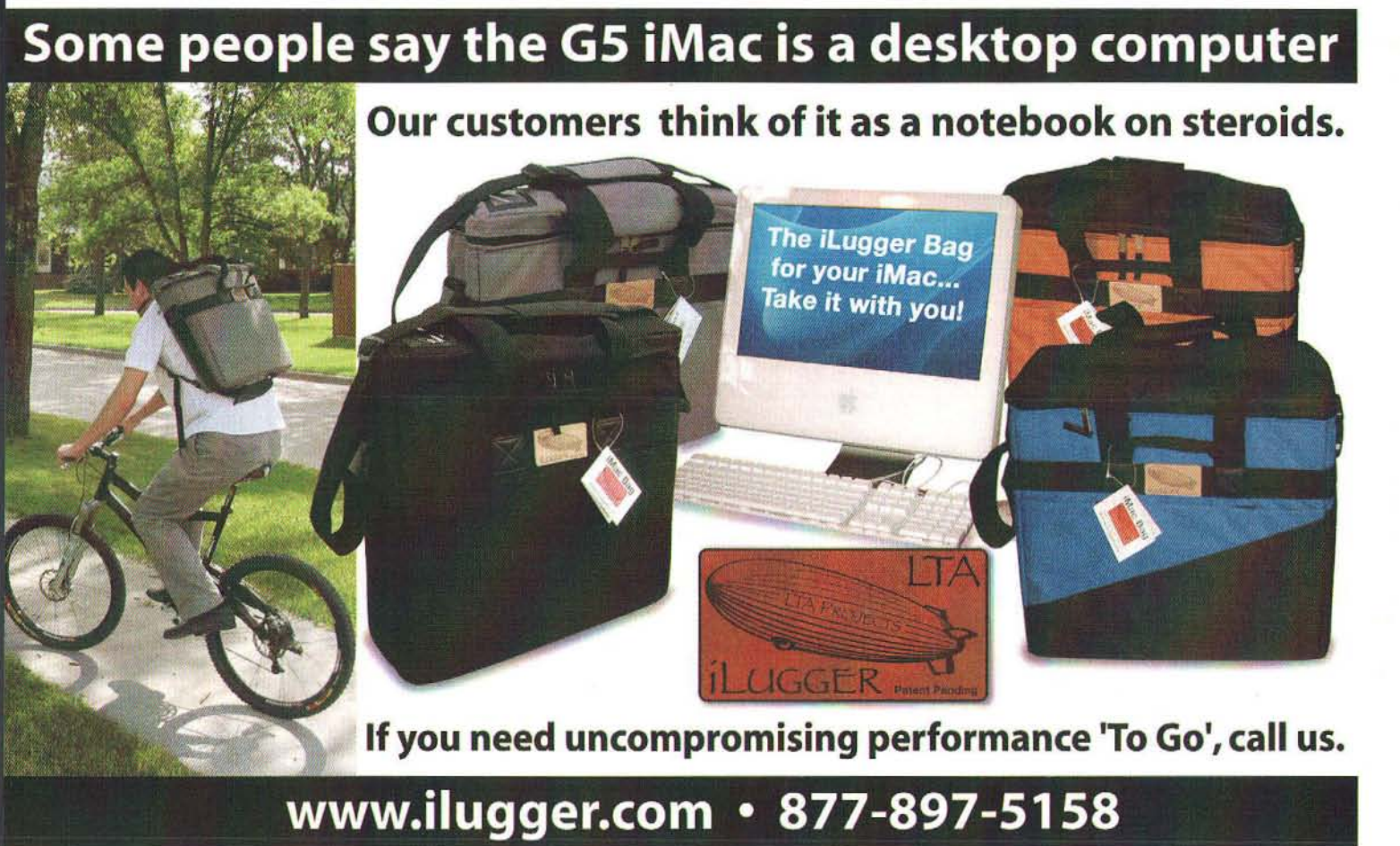
Much like cron, xinetd still exists, but is sitting idle. Also like cron, its features have been subsumed by launchd. ***Unlike*** the advice I gave for cron, this is a system to abandon on OS X.

xinetd, like inetd before it, and now launchd, all act as "superservers." Originally, every daemon on a system would launch itself, daemonize, deal with source IP checking (maybe), reserve their listening port, etc. Wietse Venema created "tcp-wrappers", inetd, to act as a master listening agent, which could be configured to enable, disable and restrict access to other daemons that needed to listen for outside requests. The request actually comes into the super-server and then, if appropriate, gets passed on to the real daemon to do the work.

Launchd really excels here: from it's deep integration into OS X, through its efficient use of resources, it's the way to go. Use xinetd if you need to wean yourself off due to use on another Unix. However, make sure it's only a transition. Switch to launchd as soon as possible for this!

Networking Tools

While the use of some of these CLI utilities are deprecated under OS X, they all still work. Certainly ***not*** deprecated is netstat. netstat shows information about various network related data structures. It's primarily used to show active network sockets. This is one way you can hunt down exactly what's listening to a port



(http://www.radiotope.com/writing/?p=8). netstat will also display the routing table when used with the '-r' switch – an important troubleshooting tool.

ifconfig has been the traditional way to view and set information about a network interface. While ifconfig does work, it may not do precisely what you want. It does still work perfectly as a way to view information about an interface. However, to change or set up an interface, ifconfig has been superceded by networksetup. networksetup is as extensive as any other tool, including any GUI, and should be used over ifconfig.

Like other Unicies, OS X sports an /etc/resolve.conf file. The resolver file configures access to DNS servers for applications, including order of name server lookup, their ports and search domains. Unlike other Unicies, OS X will automatically fill in and adjust /etc/resolv.conf based on your network configuration. Other Unices use /etc/resolv.conf as the origin of this information, and you edit it manually there. OS X plops the sum of other sources into /etc/resolv.conf. Don't edit it manually!

ipfw, the IP FireWall, from FreeBSD has also been included in OS X. It's what handles the software firewall behind the scenes. Note, though, that ipfw is a FreeBSD creation. Linux uses iptables, OpenBSD uses pf, and other Unicies tend to have their own solution.

Config Files

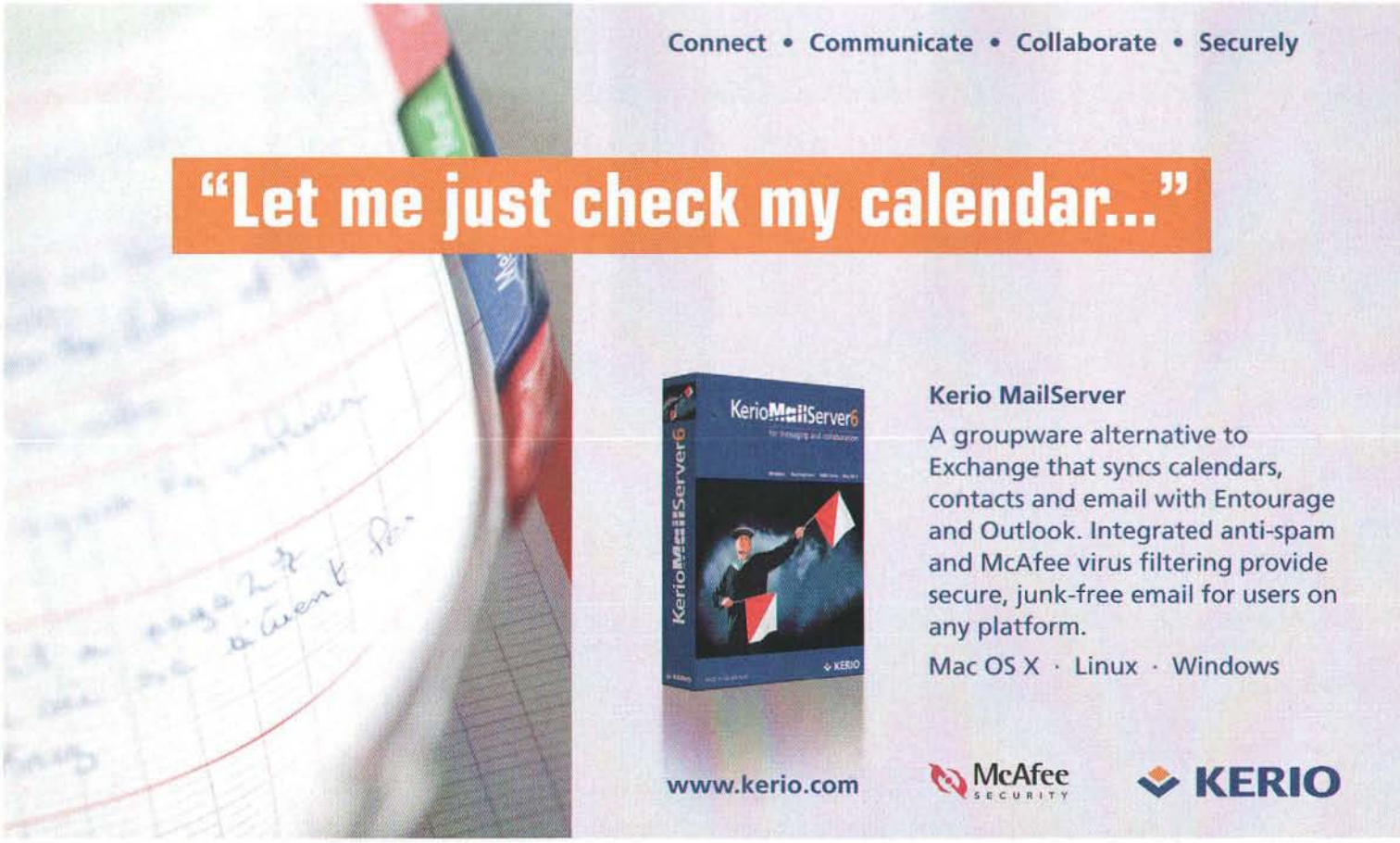
Traditionally, Unix has been a system configured by text files. While Apple has introduced non-plain text config systems, such as NetInfo, OS X does tend to respect the traditional text files for compatibility reasons.

/etc/passwd, traditionally the user authentication database, is still present on OS X. /etc/group, used for group association is also present. Both files are ignored by default, but programs that may not know any better are free to read them. You can have the system reference these files normally by selecting the "BSD and Flat Files" option in Directory Access.

/etc/hosts is present ***and*** active by default. This is actually really good news. /etc/hosts is a host to IP database that is consulted ahead of DNS. I use hosts on a daily basis as I test web sites that I'm developing on my local machine. This way, each entry can be set up as a name-based virtual host in Apache. Very handy.

Text Processing Utilities

Due to the number of text files used to configure a traditional system, Unix is awash with programming languages and utilities that manipulate text. Thankfully, Apple saw fit to include all of the current choices out of the box.



Premium

Small Business Management & Accounting

Software

Mind Your Own Business. Smarter.

800.322.MYOB (6962)

www.myob-us.com

MYOB and the MYOB logo are registered trademarks of MYOB Technology Proprietary Limited.

perl version 5.8.6 is ready for use under OS 10.4.3. This is an especially nice treat seeing Apple listen closely to their developer audience. During the Jaguar days, perl tended to languish a few revisions behind the current release. Additionally, due to some non-standard choices made for 10.1 and 10.2, updating perl was not easy. Thankfully, version 5.8.6 is close to current – close enough that most utilities won't gripe about it. (Current is 5.8.7, but there's always some script that relies on the latest!).

sed, awk and grep are all where you expect them to be. Last month's column introduced sed, and will have a part 2 next month. These three form the triumvirate of power tools for text processing. Additionally, OS X includes 'tr', the translation utility and, of course, sort – one of the oldest text utilities for Unix platforms.

To round out the programming camp, OS X includes, out of the box, Python, ruby and PHP. Kudos to Apple for keeping this diverse base of utilities on the system, developers happy, and switchers comfortable.

Development

You may have noticed that many of the aforementioned utilities, and other products on OS X are open source that started life elsewhere (including Apache, sendmail, Postfix, Cyrus and others). What makes all of this possible, aside from the source code, is having development tools loaded up. Like some other Unicies, OS X sports gcc, the GNU C compiler. There's a link from 'cc' to the current 'gcc' that acts as a compatibility layer – most Unicies had run their C compiler simply as 'cc'. Most open source software that I've had the pleasure of working with compiles cleanly on OS X. This includes the recently released MySQL 5.

Additionally, the GNU Debugger, gdb, is on the system and ready for use. Interestingly, XCode is partially just a big, convenient front-end to gcc and gdb. Choosing to use these development tools certainly makes switchers feel right at home. Of course, once you get deep into OS X development, there are some great utilities that let you get deeper into the system (like CHUD and Shark).

Mail Systems

Thanks to the open development tools and open source software mentioned in the previous section, the mail systems bundled with OS X are the same ones you can find running on many other Unix platforms. The first huge win is Postfix. Up through OS X 10.2, sendmail was the default MTA. 10.3 brought the switch to Postfix. Written from the ground up with security in mind, Postfix is also an incredible performer. While there are many MTAs in use, including the venerable sendmail (and exim and qmail...) they all operate the same no matter the platform – save Apple's extensions that hook into OS X's authentication methods, which you can find the changes of on Apple's developer site (http://developer.apple.com/darwin).

A popular and powerful POP and IMAP server is Cyrus, from Carnegie Mellon University. Also introduced to OS X as of 10.3, Cyrus has been modified by Apple to work with their custom authentication schemes.

I covered troubleshooting Cyrus/Postfix and DNS as it relates to e-mail in the July and August Mac in the Shell MacTech columns. An upcoming column will get into the details of administering a Tiger based mail system.

Permissions

Yes, Tiger brings us ACLs, however, I find that most people don't know about them, or don't feel comfortable using them (even though I covered them in the May 2005 issue of MacTech!). This keeps us relying on POSIX permissions. Well, welcome to the club! Every Unix out there uses the same set of permissions – even though they may also have something that is more flexible. Of course, sometime with flexibility comes complexity, so, the more basic POSIX permissions will be around for a good long time. Like cron, they're ingrained in the brains of sysadmins everywhere (and, of course, you can find an in-depth tutorial in a prepubescent Mac in the Shell dating February 2005).

Number 1

The number one thing we have in common with other Unix users is Love! OK, maybe not love....but community! Mac culture really is a perfect fit into the Unix world where people tend to be passionate about the tools they use and the platforms they run them on.

Fin

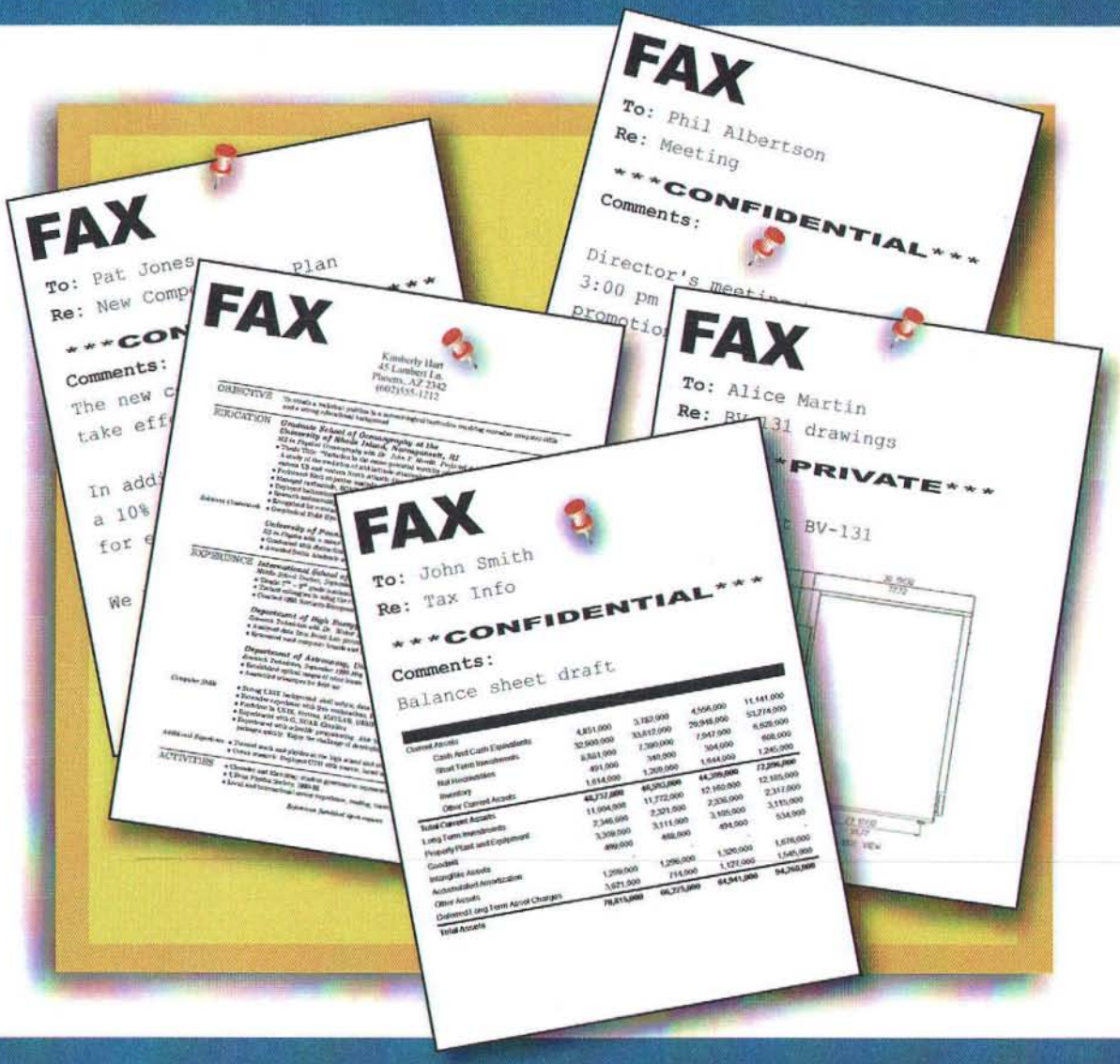
Apple *is* adding a lot with the systems they've created for OS X. Of course, it's also fantastic to be able to leverage the history that Unix brings. While I wish everyone a very happy New Year, I'll remind you again: it's a great time to assess what you know, what you want to know, and what you need to learn. When you're summing that up, don't forget one of the best systems that OS X inherits from Unix: man.

Enjoy!

About The Author

Ed Marczak owns and operates Radiotope, a technology consulting company. If you're reading this while at MacWorld Expo, and it's Wednesday, come see Ed present, "From the Chime to the Desktop" and learn all about launchd and the Mac boot sequence. Always more tech tips at http://www.radiotope.com

LOOK FAMILIAR?



Get Rid of Your Fax Machine!

Receive your faxes directly to your email account as PDF file attachments

Service available in over 150 area codes

To check out special offers for MacTech readers, visit www.MaxEmail.com/MacTech

maxemail®

ADDING AJAX TO A WEBSITE

CREATING A DYNAMIC, USER-FRIENDLY WEBSITE INTERFACE IS SIMPLE AND STRAIGHTFORWARD

BY ANDREW TURNER

Introduction

Modern websites and web-applications appear drastically different from sites on the web 5 and 10 years ago. Tools like GoogleMail, BaseCamp, and TiddlyWiki have revolutionized the general concept of what a webpage can do and how users interact with it. The days of clicking on a simple hyperlink to be taken to a new page, or sitting and waiting for a form submission are rapidly dwindling.

The technology driving these sites is not really new, but their application and use has only recently become widespread and supported by a majority of web browsers. Furthermore, many web developers feel daunted by the rapid pace of the changing techniques and don't have a clear understanding of how the technologies are implemented and used.

One of the most revolutionizing of these technologies has been dubbed AJAX (or Ajax depending upon whom you ask). Ajax is responsible for dynamic page content, marking database entries, and in-line text-editing without the need for page-reloads or large, complex plug-ins like Flash or Java.

The goal of this article is to teach you the basics of Ajax and demonstrate that it is not as difficult a concept as it may first appear. In reality, Ajax is simple and easy for any web developer to add to their new or already existing site.

What is Ajax?

AJAX is an acronym for: Asynchronous Javascript and XML. The most important concept of AJAX is the "asynchronous" part. Asynchronous communication means that commands do not need to wait for a response. By contrast, synchronous communication requires the command to wait for a response before continuing. An example of synchronous communication is a typical hyperlink; the user clicks a link, and then waits while the resulting page is requested, returned, and displayed. An asynchronous example may be having a contact name and phone number lookup with dynamic autocomplete with names already in the database. For application developers, it may be useful to think of synchronous communication as modal, while asynchronous is non-modal.

Javascript is the client-side scripting language that has been used to implement the input, output and server-response handling. Because the code is client-side it is fast and scales up with increasing usage. The last part of the AJAX acronym is XML (extensible Markup Language), which is used in the response to encapsulate information. By using a structure like XML, the client can parse the tree for specific data without having a predetermined order of the data. As we will discuss, XML or Javascript are not required to implement "Ajax" in a site.

Ajax uses several other technologies and functionality to work. XHTML and the Document Object Model (DOM) allow Javascript to dynamically modify a webpage. CSS (Cascading Style Sheets) are not necessary, but are typically employed to provide for easy layout and design of a webpage and allow the Ajax functionality to work on the data, and not the view.

The reason the technology is referred to as either AJAX or Ajax is because of the blurring between the concept, and the implementation. Ajax (non-acronym) has become the terminology associated with the ability to dynamically modify a webpage or backend content without requiring a page reload, while AJAX (acronym) is the specific implementation of Ajax employing Javascript and XML. The term Ajax was coined by Jesse Garnett of AdaptivePath (see resources) as a better name than the previously used "Asynchronous JavaScript + CSS + DOM + XMLHttpRequest". The technologies were all originally combined by Microsoft for developing their Outlook Personal Information Manager (PIM) web application interface.

Why use Ajax?

While the web has inarguably drastically changed the way a computer user works, to date they haven't been able to fully replace, or even work entirely in tandem with, desktop applications. To clarify, a desktop application is software that must be installed on a user's computer and is run in a self-contained window/context. By contrast, a web application operates primarily within a user's browser and is not required to be installed on a machine. This provides users access to the application and associated data from any computer using a suitable browser.

However, with the advent and widespread use of technologies such as Ajax, users can now complement, or even replace, their desktop applications with a web application. Many users are now switching to reading their email in GoogleMail, storing their documents and notes in TiddlyWiki, reading their RSS news via Gregarious, or working with colleagues in BaseCamp.

Adding Ajax to your own web site or web application provides a much smoother, and rich user experience. Furthermore, Ajax websites more closely imitate their desktop counterparts, allowing users to interact and understand the user interfaces in a similar way.

Ajax is also a relatively straightforward and simple technology to provide in a website. Developers may quickly become confused by all of the terms, techniques, and options. However, at its core, Ajax is quick to setup and begin using, and completely flexible for whatever the developer and site requirements need. Ajax can be used for features such as inline form validation, database queries, content editing, drag-and-drop, page updating, and many others

Setting up the framework

Parts of Asynchronous Communications

In order to understand the essential parts of an Ajax framework, we will discuss the necessary parts of asynchronous communications. The parts are split up by **Client**, the user's browser, and **Server**, the website hosting server.

Client: Create a request object
Client: Assign a response handler
Client: Send a query to the server
Server: Receive the query, and perform operations
Server: Send the response to the client
Client: Handle the server response

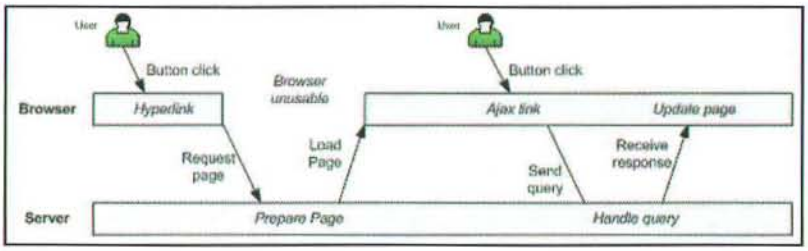
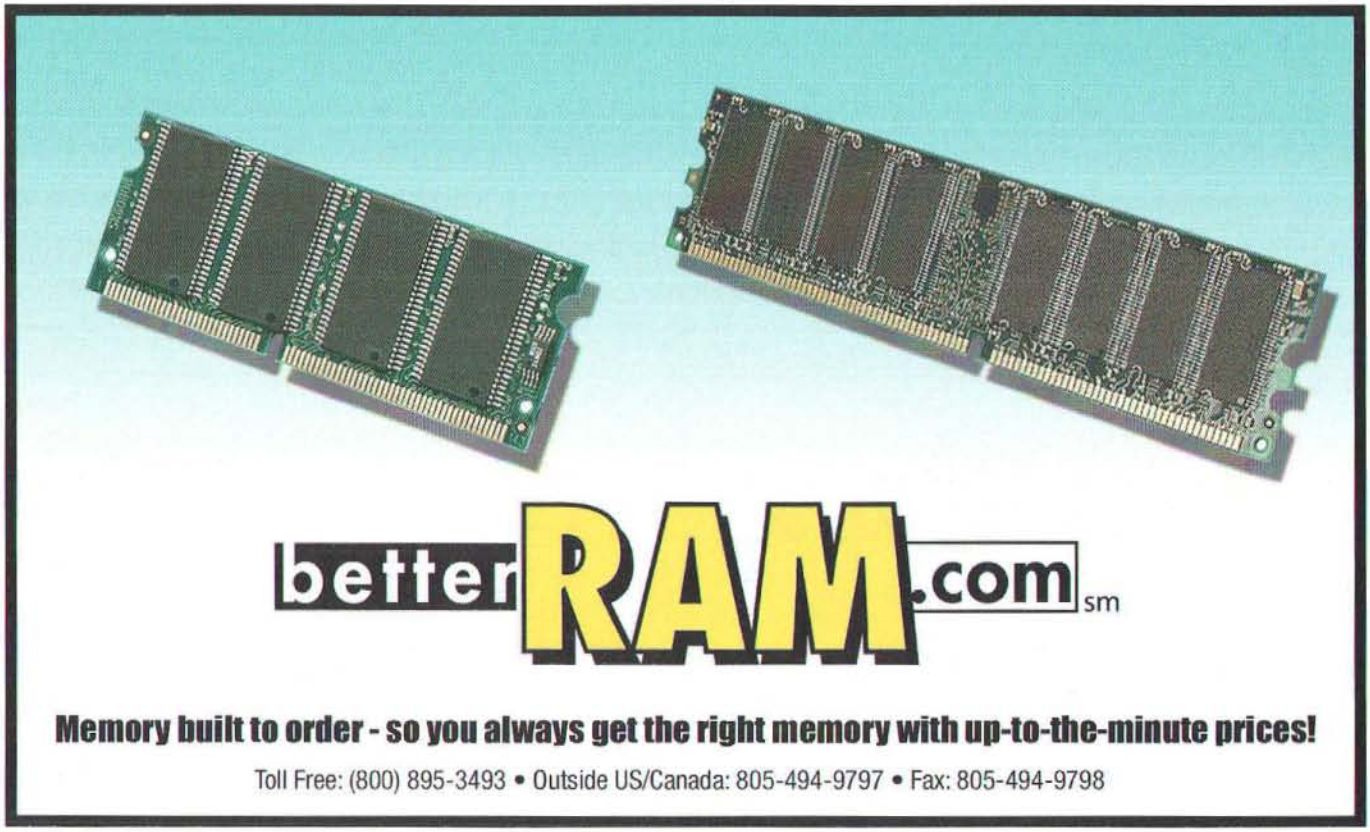


Figure 1: Asynchronous communications allow a user to continually interact with the browser, and provides dynamic updating of the web site



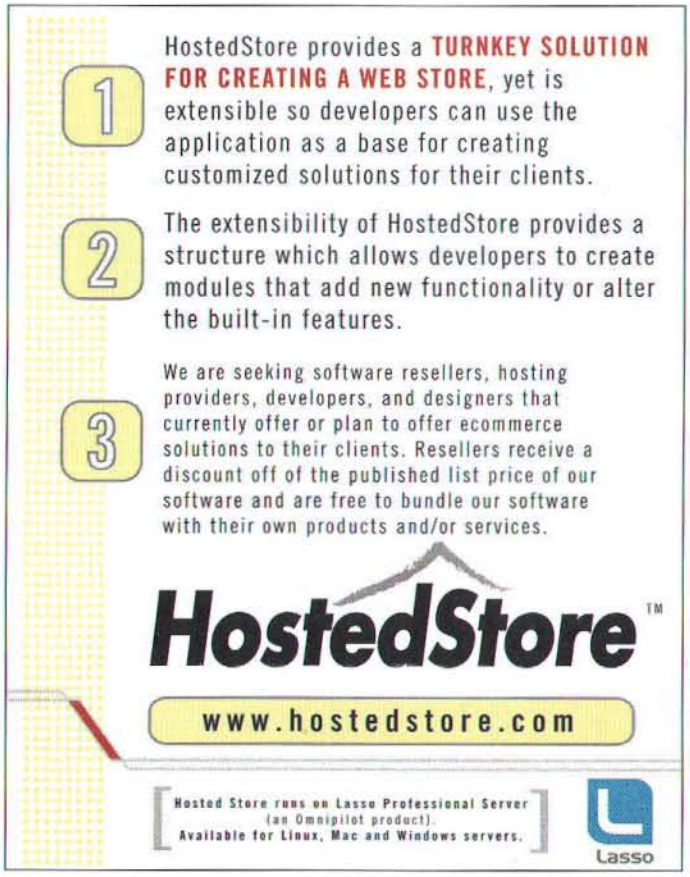
The client requests to the server can happen continually, updating the web page or application on each response received. Each request happens separately from the interface, allowing the user to continue to view and interact with the web page.

However, it is a good idea to only support a single asynchronous command at a time as the response may affect the interface data. If multiple asynchronous requests are supported, you must be careful to handle potential conflicts due to user interaction with outdated data.

Create a request object

The first thing to do is to create a *constructor* that will build a client-side request object. A request object is responsible for wrapping up the actual request, response handler, and state of the request.

Remember that this Javascript code is being run on the user's desktop browser. Therefore creating the request object is the one place where browser specific code is required. In this case, Microsoft's Internet Explorer uses an ActiveX object as the request object, whereas the other browsers all support an *XMLHttpRequest()* constructor call. We can interrogate the browser to find out what type it is and create the appropriate object. This function is universal for any Ajax use.



AjaxFramework.js

```
// request object constructor
function createRequestObject() {
  var ro;
  var browser = navigator.appName;
  if(browser == "Microsoft Internet Explorer"){
    ro = new ActiveXObject("Microsoft.XMLHTTP");
  }else{
    ro = new XMLHttpRequest();
  }
  return ro;
}
```

You should now create a global request object that will be used by the client for all future communication.

AjaxFramework.js

```
// global request object
var http = createRequestObject();
```

Assign a response handler and handle the response

Our second step is to assign a response handler. A handler is the function that will be called when the request comes back from the server to the client's computer. This function is responsible for verifying the state of the answer, and parsing the response as appropriate. This function is implemented on a project specific basis. It needs to know what the expected response from the server looks like and how to place that response back into the user's browser document.

AjaxFramework.js

```
// callback function that handles any state changes of our
request to the server
function handleResponse() {
  if(http.readyState == 1){
 // request loading
      document.getElementById("status").innerHTML
        = "requesting...";
  }
  else if(http.readyState == 4) {
// request complete
    if(http.status == 200) {
// OK returned
      var response = http.responseText;

      // Add more advanced parsing here if desired

document.getElementById("responseArea").innerHTML
        = response;
    }
    else
    {
      document.getElementById("status").innerHTML
        = "error: " + http.statusText;
    }
  }
}
```

The first thing the handleResponse function does is check the current state of the request object. If the object is loading (1), then the user is alerted to this, or if the request is complete (4) then we handle the response. This

Multiple Formats. Multiple Platforms. Complex Installers.

We have the solution:

StuffIt Engine SDK

Solving the compression & multi-platform puzzle!
www.stuffit.com/sdk/

Put the power of StuffIt to work for you.

- Adds value to your applications by integrating compression and encryption tools.
- The only tool that supports the StuffIt file format.
- Make self extracting archives for Macintosh or Windows
- Available for Macintosh, Windows, Linux or Solaris

Licenses start as low as $99/Yr.

Mac

solaris

StuffIt InstallerMaker

An OS X native version ready for developers!
www.stuffit.com/installermaker/

Give your software a solid beginning

- Create Macintosh OS X and Macintosh Classic compatible installers
- Get all the tools you need to install, uninstall or update your software in one complete package
- Add muscle to your installers by customizing your electric registration form to include surveys and special offers.

Prices start at $250

Allume Systems
www.allume.com
email: dev.sales@allume.com

2004 Allume Systems, Inc. StuffIt, StuffIt Installermaker and StuffIt Engine SDK are trademarks or registered trademarks of Allume Systems, Inc. The Allume logo is a registered trademark of Allume Systems. All other products are trademarks or registered trademarks of their respective holders. All rights reserved.

example just puts the response text (use ***responseXML*** for an XML response from a server) into our document's ***responseArea***.

Send a query to the server

Now that we have setup the request object structure, as well as the state handling function, the next step is to create a function that our webpage will be calling for each outgoing request. This function could either accept information via an input parameter, or retrieve user input by querying the document.

Once the user input is received, we create a ***GET*** request to a URL. It is important to note that due to security concerns the request can only be made to a server that is hosting the webpage. The domain name must be exactly the same as the request URL, if there is a preceding ***www.*** to the domain name. As usual, however, there are some fairly straightforward work arounds for getting external data for your Ajax requests. Several options will be discussed.

This example demonstrates using a REST input (parameters passed via the URL), but other remote query and command options are also possible. Furthermore, the ***open*** command supports passing a username and password to the server for accessing protected services.

AjaxFramework.js

```
// function for filling out and sending a request - called by the
actual webpage
function sendRequest() {
  var query =
document.getElementById("queryInput").value;
  var queryURL =
"http://localhost.com/service.php?q=" + query;
  http.open('GET', queryURL);
  http.onreadystatechange = handleResponse;
  http.send(null);
  return true;
}
```

We have now completed the necessary parts of our Javascript code to handle creating, sending, and receiving an asynchronous request through a client's browser.

Server handling of the request

The client makes a request to some service or page that is served on the same domain as the original webpage. This service for this example is expecting a value passed via the URL in the GET parameters. The response can be well formed XML, or simple text that will be parsed by the client's browser as discussed above in the ***handleResponse()*** function.

service.php

```
<?php
$query = $_GET['q'];
$response = some_service_handling($query);
echo $response;
?>
```

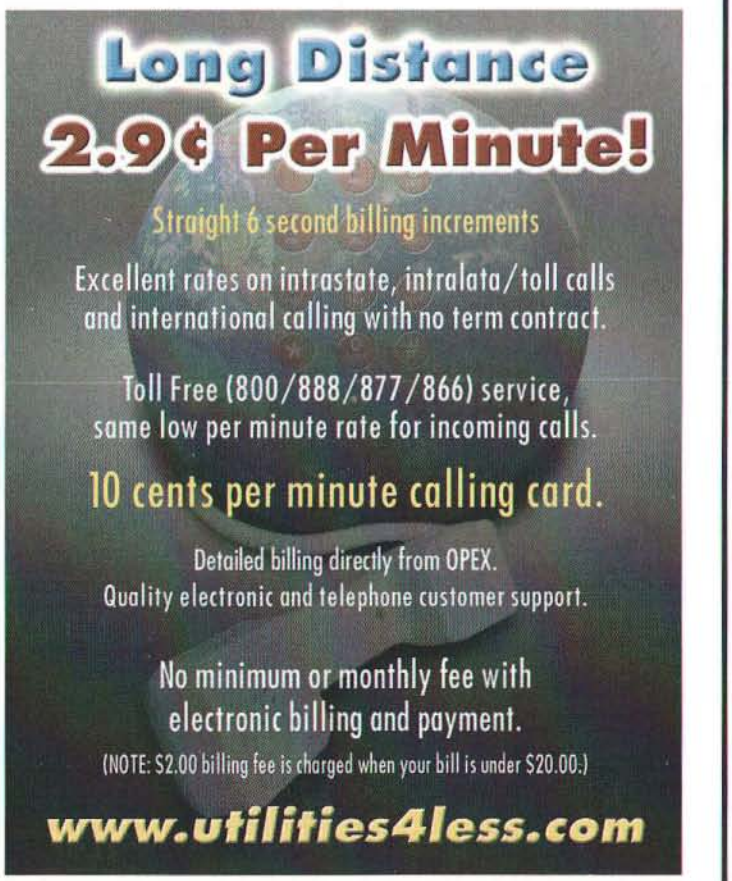
This server page just passes the query onto another php function and then echoes the response. Since our Ajax request from the browser has made a ***GET*** request, this operates like any normal opening a page in a browser. However, instead of the page showing up in a window, it is handled by the client's ***handleReponse()*** function.

Using a remote service

As we mentioned earlier, security does not allow the Ajax, specifically XMLHttpRequest, to call another domain in the GET URL. The way around this is provide a locally served wrapper to the remote service. We can parse and pass on each of the incoming parameters. Also, many hosting services don't allow a URL to be opened via the ***fopen()*** command, so this example uses ***curl*** to make a request to a server. The subsequent response is read by the local server and then returned to the calling Ajax function.

remote_service.php

```
<?php
$remote_params = "";
foreach($_GET as $key=>$value)
{
  $$key = $value;
    if($value != '')
        $remote_params .= "&".$key."=".$value;
}
$remote_url =
"http://remotehost.com/remoteservice.php?";

function get_content($url)
{
    $ch = curl_init();
    curl_setopt ($ch, CURLOPT_URL, $url);
```

no more auction management headaches.

INDICATIONS: For rapid relief from monthly fees, disorganization, and lost sales. Aids in listing, inventory management, email marketing, trends analysis, customer retention management, and other activities associated with selling profitably on eBay.

DOSAGE: One installation. Subsequent updates are automatic.

DIRECTIONS: List. Sell. Repeat.

ACTIVE INGREDIENTS: automation. scheduled events. custom actions. offline capability. import from turbolister, excel, and databases. ledger. groups. cash flow analysis. support for eBay stores and eBay motors. custom reporting. html editor. bulk listing, ending, revision. relisting. free scheduled listing. offline listing preview. ad templates. image management. watermarks. spell check. consignment management. profiles. vendor management. auto reorder. product sales analysis. questions and feedback management. spam control. fraud management. email templates. invoicing. dispute management. email marketing campaigns.

WARNING: Side effects may include increased sales, higher efficiency, and euphoria.

marketblast

please visit www.marketblast.com for a free sample.

©2005 4D, Inc. All rights reserved. MarketBlast, related logos and all 4D product names are registered trademarks of 4D Inc/4D SA.
All other tradenames or registered tradenames are trademarks or registered trademarks of their respective holders.
MarketBlast is not an actual pharmaceutical product. Do not ingest.

```
    curl_setopt ($ch, CURLOPT_HEADER, 0);

    ob_start();

    curl_exec ($ch);
    curl_close ($ch);
    $string = ob_get_contents();
    ob_end_clean();

    return $string;
}

$content = get_content
($remote_url.$remote_params);
echo $content;
?>
```

This example makes no checks on the incoming request. The query and parameters are passed directly onto the remote service. In a real application, it would be responsible to do some basic parameter checking before passing on the request to someone else's hosting service.

That said, it is still a means by which to provide asynchronous services in your own website. You should also be aware that making these remote calls may have longer response times. While this situation is an excellent reason why an asynchronous interface provides a better user experience, users may be left wondering if their request was just lost. Therefore, you should, when appropriate, let the user know that the request is pending in some way.

Furthermore, it would be possible to setup a timeout timer for each request that would call *abort()* on the request object if the request took too long.

Supporting non-Javascript functionality

This framework will generally work for any modern, Javascript capable browser. However, not all users are using Javascript capable browsers, and other users may have disabled Javascript. Therefore, it is advised that your site support a non-Javascript version of your interface. At the very least, alert the user that they will not be able to use all of the functionality of your web application or page.

To provide a non-Javascript interface only when necessary, your page should use the *<noscript>* tags paired with any *<script>* sections.

Using the Framework

The Javascript framework is logical backend functionality of an Ajax enabled website. In order to use Ajax, the page must be properly constructed and typically web developers also wan the page to look nice. For both of these requirements, we will use XHMTL and CSS respectively.

Example query and response

Lets illustrate the Ajax framework with an example. Our service could return a name and phone number of a contact from our webserver. The query parameter, *q*, could be some search term, and the response would be the contact's name, and phone number. Testing such a service is easy:

THE FUTURE IS CALLING.

AND IT'S USING A POLYCOM VOIP PHONE.

The Power of a Network. The Simplicity of a Phone. Polycom® is a leading independent supplier of standards-based Voice over IP endpoints. As Polycom VoIP phones are fully interoperable with leading IP PBX and Softswitch platforms, you can have a variety of end-to-end, fully integrated VoIP solutions to choose from.

Polycom Offers the Right VoIP Endpoints to Meet Your Business Communication Needs. Polycom provides an assortment of VoIP phone models plus a choice of supported protocols and call management solutions to match the unique business communication requirements of your organization. The Polycom VoIP product portfolio includes the SoundPoint® IP family of desktop phones, an attendant console based on the SoundPoint IP 601 and Expansion Module, and the SoundStation® IP 4000 conference phone. Our phones are engineered to deliver a superb communications experience. They offer outstanding sound quality, advanced functionality, ease of use, simplified configuration and upgrades, and protection of your investment over time.

Please visit the Polycom Experience Center at www.polycom.com/experience to learn more about our VoIP products and other Polycom solutions to enhance your business communication needs.

POLYCOM®

VIDEO VOICE DATA WEB

TOGETHER, GREAT THINGS HAPPEN.

http://localhost.com/service.php?q=Jones

We will then expect a response like:

Edward Jones, 800-555-1212

Our ***handleResponse*** functionality need to be expanded to ***split*** the response with the comma (or multiple commas for more data).

AjaxFramework.js (handleResponse)

```
var data = response.split(',');

// more advanced parsing
  document.getElementById("contact_name").innerHTML
        = data[0];

document.getElementById("contact_number").innerHTML
        = data[1];
```

This example response illustrates how easy Ajax is to begin to use. There is no need for complex XML parsing and handling. Any simple response can be used to dynamically update web content. We must be careful, however, as we have forced the response to return the information in a specific order and format.

A more robust application should use XML to provide multiple contact data. The response handler could then iterate through the elements of the contact entry without having to predetermine the order of the response. In our example, if we switched the contact name and contact number order the application would behave incorrectly.

In this case, we would instead get the ***responseXML*** and parse the XML document tree similar to the DOM of the browser document.

AjaxFramework.js

```
var response = http.responseXML;
var contact_name =
  response.getElementsByTagName('name').item(0);
var contact_number =
  response.getElementsByTagName('number').item(0);
```

However, XML is not necessary, and may be daunting when first starting to use Ajax, or integrate it into already existing web services. Therefore, we will continue our example using the simple text response. Since the XML handling is encapsulated in the ***handleResponse()*** function, it is possible to later change to using XML without modifying the rest of our framework.

Example page

To use the Ajax searching, we will need to provide an XHTML user interface for the query input, and the service response. The first thing we need to do is include our Javascript framework code in our page:

```
<html>
<head>
<script type="text/javascript"
src="AjaxFramework.js" charset="utf-8"></script>
</head>
```

Next we create the query input and "Send" button. Note the use of the ambiguous anchor link, #, in the ***bref*** tag. We use an ***bref*** link to allow standard style formatting of the "Send" button to match the rest of the sites hyperlinks. By using the local anchor, but with no actual anchor, the hyperlink won't cause a page refresh since the browser thinks it is just scrolling down the current page. Another option would have been to use a generic div and provide a unique formatting for Ajax link as compared to actual hyperlinks.

```
<body>
<input type="text" size="30" id="queryInput"
value="" />
<a href='#' onClick="sendRequest();">Send</a>

<div id="status"> </div><br/>

<div id="contact_name"> </div>
<div id="contact_number"> </div>
</body>
</html>
```

When the "Send" link is pressed, the ***queryInput*** text input is sent as a query to our name lookup service. The user is free to continue to use the web browser. When the response is sent from the server, the retrieved name and number are placed in the ***contact_name*** and ***contact_number*** divs.

A more advanced version of this application could add in-line searching of the contact name as the user types, similar to autocomplete.

Summary

Ajax is quickly transforming websites from repositories of data into dynamic and useful web applications. This article demonstrated how easy it is to get started with Ajax and add it to your own site. Some examples you can use it for include form checking while the user is entering information, site/document search, database row updating, or editing web content in place.

For more advanced applications you may want to look at several available and supported Ajax toolsets that provide a ready framework and lots of other functionality. Prototype (see resources) is used in Ruby on Rails for its Javascript Ajax functionality, and Sajax is an Ajax toolset for PHP code.

Resources

Ajax technology

Adaptive Path:

PROTECT YOUR POD

These lightweight super-tough aluminum cases for the iPod® are the most durable and stylish EVER!

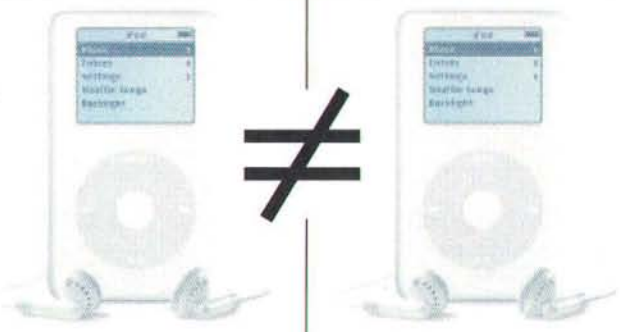
iArmor™ iKeychain™

Customized engraving and other colors available.

www.A1QP.com

Designed and manufactured in the USA by A-1 Quality Products, Inc.
©2005 All rights reserved. iArmor and iKeychain are trademarks of A-1 Quality Products, Inc.
iPod is a registered trademark of Apple Computer, Inc.

What's the *difference*?



Brand new. It's perfect. It's beautiful...

and it's *vulnerable*!

It's naked, exposed to the elements.

It's...

INHUMANE

and it *will* get scratched.

Brand new. It's perfect. It's beautiful...

and it's **protected**!

Invisible, durable, & rugged protection.

SAVE the iPod

www.the*invisible*SHIELD.com

Sidewinder™

www.MARWARE.com

Macworld Conference & Expo
Booth # 2035

You didn't compromise on your music player—
don't compromise on your case.

kickstand for video viewing

http://www.adaptivepath.com/publications/essays/archives/000385.php
Mozilla Ajax documentation:
http://developer.mozilla.org/en/docs/AJAX
XMLHttpRequest documentation
http://documentation:
developer.apple.com/internet/webcontent/xmlhttpreq.html

Ajax applications

TiddlyWiki: http://www.tiddlywiki.com
Gregarius RSS Reader: http://gregarius.net
Basecamp: http://www.basecamphq.com
Geocode lookup (with source):
http://highearthorbit.com/projects/geocode/

Ajax toolsets

Prototype, a Javascript framework for web apps: http://prototype.conio.net/
Sajax, Ajax for PHP: http://www.modernmethod.com/sajax/

Ajax Framework

The following files are the summation of the framework code developed in the article above. It can serve as a skeleton for building your own Ajax applications. Place these files in your /Library/WebServer/Documents directory on your Mac, and turn on "Personal Web Sharing" in the "Sharing Preference Pane".

AjaxFramework.js

```
function createRequestObject() {
  var ro;
  var browser = navigator.appName;
  if(browser == "Microsoft Internet Explorer"){
    ro = new ActiveXObject("Microsoft.XMLHTTP");
  }else{
    ro = new XMLHttpRequest();
  }
  return ro;
}
var http = createRequestObject();

function handleResponse() {
  if(http.readyState == 1){

 // request loading
      document.getElementById("status").innerHTML
        = "requesting...";
  }
  else if(http.readyState == 4) {

// request complete

    if(http.status == 200) {

 // OK returned
      var response = http.responseText;

      document.getElementById("status").innerHTML
        = "loaded";

document.getElementById("responseArea").innerHTML
        = response;
    }
    else
    {
      document.getElementById("status").innerHTML
        = "error: " + http.statusText;
    }
  }

}

function sendRequest() {
  var query =
document.getElementById("queryInput").value;
  var queryURL = "service.php?q=" + query;
  http.open('get', queryURL);
  http.onreadystatechange = handleResponse;
  http.send(null);
  return true;
}
```

AjaxDemo.html

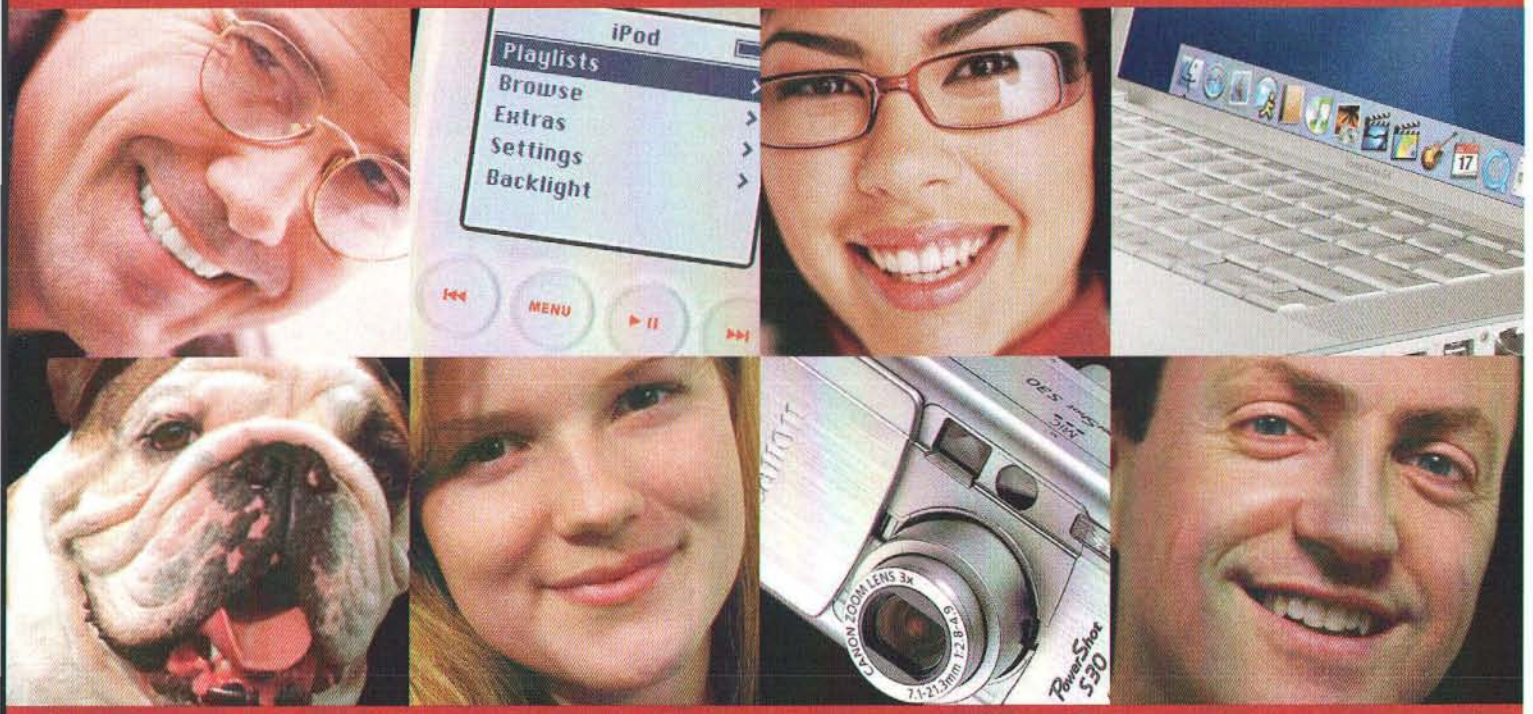
```
<html>
<head>
<script type="text/javascript"
src="AjaxFramework.js"></script>
</head>
<body>
<noscript>
Your browser does not support Javascript. Please
upgrade your browser or enable Javascript to use
this site.
</noscript>

<input type="text" size="30" id="queryInput"
value="" />
<a href='#' onClick="sendRequest();">Send</a>

<div id="status"> </div><br/>

<textarea rows="20" cols="70" id="responseArea"
value="" ></textarea>

</body>
</html>

service.php
<?php
echo $_GET["q"];
?>
```


About The Author

Andrew Turner is a Systems Development Engineer with Realtime Technologies, Inc. (www.simcreator.com) and has built robotic airships, automated his house, designed spacecraft, and in general looks for any excuse to hack together cool technology. You can read more about his projects at www.highearthorbit.com.

MACTECH®
Get MacTech delivered to your door at a price FAR BELOW the newstand price. And, it's RISK FREE!
store.mactech.com/riskfree

At Small Dog Electronics, happy customers are our highest priority.



When you shop at Small Dog Electronics, you get more than just great selection and low prices; you also get personalized service from genuine Apple Professionals who take customer service very seriously. And that's a promise... **no if's, and's or but's.**

Small Dog Electronics
www.smalldog.com

1-800-511-MACS

Apple Specialist

A socially responsible business since 1996

Choosing A Java Scripting Language

If you are considering hooking a scripting interpreter into your Java application, the hardest part is choosing which one to use.

By Paul T. Ammann

Scripting languages have proven their value to Java developers. They let users extend and customize the application's functionality and appearance, which adds value to the program. In addition, they can dramatically simplify certain design tasks for a programmer by making it possible to define, load, and evaluate functions on the fly. For a developer, the task of integrating one or more of these scripting languages is easy. Picking one from the growing list is hard. This article describes some of the issues that come with supporting a scripting language in your Java application and compares Groovy, JudoScript, Pnuts, JRuby, Jacl, Jython, Rhino, and BeanShell in a variety of ways to help you make the right choice.

A couple of years ago, I needed a Java scripting language and tried to choose one that seemed like a good fit for a demanding commercial application. Ideally, I wanted an interpreter that would ease the extension of the application's user interface, have readable scripting code, be reliable, fast, well-supported, well-documented, and complete. At that time, I had narrowed my choice down to Jacl, Jython, Rhino, and BeanShell.

A lot has changed since that time. Instead of a handful of choices, there are now more than a dozen scripting languages either under active development or already available for use. The list of solid choices is bigger than it was years ago and now includes Groovy, JudoScript, Pnuts, and JRuby, in addition to Jacl, Jython, Rhino, and BeanShell. We could consider other scripting interpreters beyond this group, but this list is large enough for developers to find what they're looking for.

I wanted to benchmark all of these interpreters to see if the performance for Jacl, Jython, Rhino, and BeanShell has improved since 2002, and to find out how Groovy, JudoScript, JRuby, and Pnuts compare with them. I thought it might be interesting to see what's unqiue about the different scripting interpreters and if any have particular strengths or weakness.

Risky Business

The benefits of Java scripting interpreters are substantial. For one thing, scripting languages can be simpler to code in than Java. Scripts also make it possible to drive and extend your program's application logic and user interface. They can be run directly against your Java application's class interfaces, which is very powerful. This can make it easier to write test drivers against your program much more quickly than if you had to code and compile unit tests against the Java classes themselves. Also, if users take the time to extend your application by writing scripts for it, they're making an investment in your tool—and that can give your application an edge against the competition.

You do open yourself up to a certain amount of risk by integrating a Java scripting interpreter into your application, though. The two biggest risks are that the interpreter will be orphaned or

that you will discover a fatal flaw in the interpreter after you ship a product with it.

Most of the interpreters are actively maintained and updated through an open source model, and, in those cases, you can probably rely on experts for help on working around problems you find, patching the interpreter, or getting the bug-fix you need included in a future release. That's a safe bet, but not a guarantee. If you are seriously considering using a specific interpreter, take a look at the activity on the development site to get a feel for how the code is evolving and look at the message boards to see if user questions are getting answered. That will give you a feel for how well supported the code really is.

Another way to protect yourself is to thoroughly test any scripting interpreter you plan to use. The distributions for some interpreters include a set of unit tests. When you test the scripting interpreter integration with your application, those unit tests can serve as part of the larger test suite you'll want to put together. When you test the integration between the interpreter and the application, you have your work cut out for you, because scripting interpreters are so flexible and expose so much functionality to the developer. You're making an investment by spending time on quality assurance early on, instead of when the application is in production or when customers need a critical bug fixed.

Scripting language	Version number	Description
Jacl	1.3.1	The Java implementation of the Tcl interpreter. If you want to use Tk constructs in your scripts to create user interface objects, take a look at the Swank project for a set of Java classes that wrap Java's Swing widgets into Tk. Jacl has been around for quite a while and is still being actively worked on.
Jython	2.1	The Java implementation of the Python interpreter. One concern I have about this interpreter is that a new release has not been issued in quite a while. However, there are signs on the Jython Website indicating plans to reverse this trend, apparently with funding to back up the work.
Rhino	1.6.1	The Java implementation of the JavaScript interpreter. It also supports compiling the scripts into classfiles. The most recent Rhino release came out just a few months ago and added support for XML.
JRuby	0.8	The Java implementation of the Ruby interpreter. JRuby is under active development. I tested version 0.8, which worked well.
BeanShell	2.0 beta 2	BeanShell is a Java source interpreter. BeanShell continues to evolve and add new features. Version 2.0 provides full support for interpreting ordinary Java source code.
Groovy	1.0 beta 9	There actually are some groovy things about this new language that add some of the features of Python and Ruby to a Java-like syntax. You can compile the scripts directly into Java classfiles. Groovy plug-ins are available for numerous IDEs, and a JSR (Java Specification Request) committee is working on the language specification for Groovy.
JudoScript	0.9	JudoScript has a JavaScript-like programming syntax that is easy to learn and use. One of the stated goals mentioned in the JudoScript FAQ is to "support object-level, OS-level, and application-level scripting." I tested version 0.9, which worked well.
Pnuts	1.1 beta 2	Pnuts has a Java-like programming syntax and seems to be actively developed and released. You can compile the scripts directly into Java classfiles.

The New List of Contenders

If you're looking for a scripting interpreter, there are plenty to choose from. Some interpreters have been written to support existing languages like Ruby, Python, JavaScript, Java, and Tcl. Other interpreters, like JudoScript, Groovy, and Pnuts, have chosen their own language syntax that is similar to Java but, in some ways, different. One of the biggest choices you'll have to make when you compare different interpreters is deciding what scripting language syntax is a good fit for you. Technology choices like this one, where developers' personal preferences come into play, can spark heated discussions among software development teams. Perhaps the information in this article can help settle some arguments.

I collected the most recent releases of eight different scripting interpreters for comparison. The interpreters and their version numbers are listed on the previous page. (See table) If you are not familiar with these interpreters, I've also included a thumbnail sketch of the functionality and development activity on each one below.

The First Benchmark: Performance

For the first benchmark, I wrote equivalent scripts for each of the interpreters to do a set of simple tasks and then timed how long it took each interpreter to run the scripts. My benchmark scripts stick to basic operations like looping, comparing integers against lots of other integers, and allocating and initializing large one- and two-dimensional arrays. The benchmarking scripts for each of the languages and the Java programs to run them can be downloaded from resources referenced throughout this article.

The most useful information that comes out of the benchmarking tests is an apples-to-apples comparison of how quickly the interpreters complete some extremely simple tasks. If throughput is a high priority for you, then the benchmarking numbers are useful.

I tried to code each test as similarly as possible for each of the scripting languages. The tests were run using Java 1.4.2 on Apple's 1.42GHz PowerPC G4 and 512 MB of RAM. I used the default heap size when invoking the Java Virtual Machine.

In the interest of giving you some perspective for how fast or slow these numbers are, I also coded the test cases in Java and ran them using Java 1.4.2.

Here are the four performance tests:

- Count from 1 to 1 million
- Compare 1 million integers for equality
- Allocate and initialize a 100,000 element array
- Allocate and initialize a 500-by-500 element array

Total Time for the Four Tasks

Since several of the interpreters closely resembled each other in terms of speed (at least for my benchmarks), I summed up the times for the interpreters on the four benchmark tasks and show the cumulative results in Figure 1.

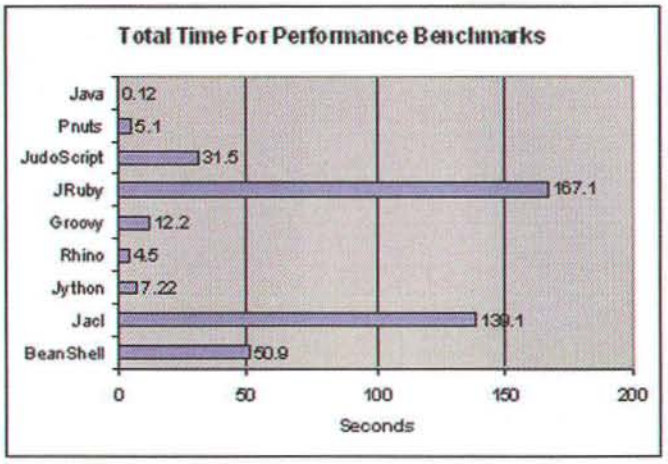


Figure 1

MACTECH® Printer Supplies

Get OEM printer ink, and toner cartridge quality at a fraction of the cost !!!

Inkjet and Laser Toner Cartridges That Create Professional Looking Documents.

We offer high quality 100% compatible inkjet cartridges for Epson, Canon, Brother, and Xerox printers. Black inkjet cartridges from $3.99, color cartridges from $5.99. We also offer Genuine OEM HP, Canon, Lexmark, Sharp, and Xerox printer ink cartridges and laser toner cartridges at discount prices.

http://www.mactech.com/supplies/

The Checkered Flag

For these simple test cases, Rhino, Pnuts, and Jython were consistently the fastest, followed closely by Groovy, then JudoScript, and the others.

Whether these performance numbers matter to you depends on the kind of things you want to do with your scripting language. If you have many hundreds of thousands of iterations to perform in a scripting function and users of your application will be waiting on the result, then you might want to either focus your attention on the interpreters at the fast end of the spectrum, or plan on implementing your most demanding algorithms in Java code instead of in scripting code. If your scripts have few repetitive functions to run, then the relative differences in speeds between these interpreters is a lot less important. A faster computer can also make a big difference in these numbers.

Another thing worth pointing out is that even the fastest of the interpreters takes about 40 times as long to run these simple programs as compiled Java code does. If speed is really at a premium for you, you might decide that it makes more sense to code certain algorithms in Java instead of using scripting code.

Some scripting interpreters support the compilation of scripts directly down to bytecode. I was curious about how much of a performance difference this would make, so I tried another test. I used the script compiler for Rhino to turn the benchmark scripts into bytecode. Then I reran the whole benchmark suite 10 times using scripts and 10 times using scripts converted to bytecode. Surprisingly, compiling the scripts to bytecode only shaved about 10 percent off the time it took to run the four programs in the benchmark suite. I initially thought that the JVM invocation must be taking the lion's share of the time to run the benchmarks, but further examination showed that the invocation of the JVM itself only accounted for about 20 percent of the total time required to run the suite. It seems that compilation of simple scripts makes a positive difference, but isn't necessarily a silver bullet for dramatically improving performance. Perhaps with longer or more compute-intensive scripts, you would see different results.

The Second Benchmark: Integration Difficulty

The integration benchmark covers two tasks. One task shows how much code it takes to instantiate the scripting language interpreter and run a scripting file. The name of the script to run is passed in as a command line argument to the ScriptRunner class. This yields a straightforward but useful program for testing scripts. Most distributions for the interpreters include much nicer console applications for interactive testing of scripts. I wanted to write a simple program from scratch to see if the documentation made using the interpreter easy or hard.

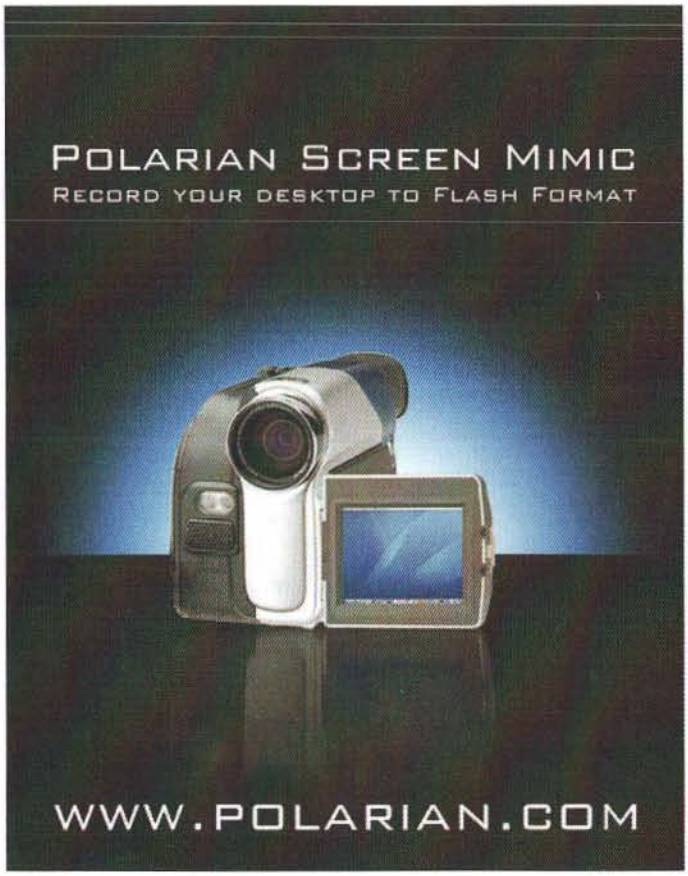
The second task writes a script that instantiates a Java `JFrame`, populates it with a `JTree`, and displays the `JFrame`.

These tasks are simple but have some value since they show how easy it is to get started using the interpreters and also what a script written for the interpreter will look like when you use it to call Java class code. I present these examples as just one way of getting started. They aren't meant to be bulletproof or even particularly complete; they provide just the essentials to get something working in that scripting language. Once you have that going, you can really start investigating the features important to your application.

Groovy

To integrate Groovy into your Java application, you create a Binding and instantiate a GroovyShell on that Binding. Then you ask the interpreter to evaluate the source at the filepath you provided on the command line. Here's what the code looks like:

```
import groovy.lang.GroovyShell;
import groovy.lang.Binding;
import groovy.lang.Closure;
import java.io.File;

public class ScriptRunner {
   public static void main (String[] args)
throws Exception {

      GroovyShell interp = new GroovyShell(new
Binding());
      try {
          File f = new File(args[0]);
          interp.evaluate(f);
      } catch(Exception e) {
          System.out.println("Exception while
sourcing file " + args[0]);
          e.printStackTrace();
      }
   }
}
```



The Groovy script to create a JTree, put it in a JFrame, and show the JFrame looks like this:

```
import javax.swing.JFrame
import javax.swing.JTree
import javax.swing.WindowConstants

class SimplestGUI {
    void buildIt() {
        frame = new JFrame("Simplest GUI");

frame.setDefaultCloseOperation(WindowConstants.DI
SPOSE_ON_CLOSE);
        tree = new JTree();
        frame.getContentPane().add(tree);
        frame.pack();
        frame.show();
    }

    static void main(args) {
        b = new SimplestGUI()
        b.buildIt()
    }
}
```

JudoScript

To integrate JudoScript into your Java application, you create a JudoEngine and then ask that JudoEngine to evaluate the file at the path specified:

```
import com.judoscript.JudoEngine;

public class ScriptRunner {
   public static void main (String[] args)
throws Exception {
       JudoEngine je = new JudoEngine();
       try {
           je.runScript(args[0], args, null);
       } catch(Exception e) {
           System.out.println("Exception while
sourcing file " + args[0]);
           e.printStackTrace();
       }
   }
}
```

The JudoScript script to create a JTree, put it in a JFrame, and show the JFrame looks like this:

```
const #JFrame = java::javax.swing.JFrame;
const #JTree = java::javax.swing.JTree;
frame = new java::#JFrame('Simple JudoScript
GUI');
tree = new java::#JTree();
frame.getContentPane().add(tree);
frame.pack();
frame.setVisible(true);

gui::events {
  <frame : Window : windowClosing>:  exit 0;
}
```

Pnuts

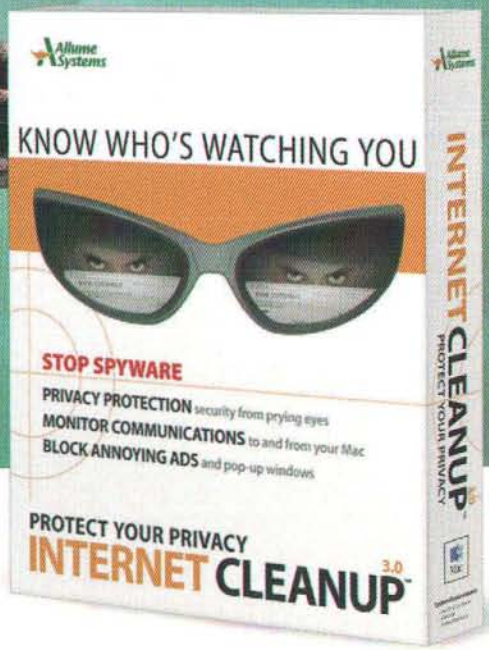
To integrate Pnuts into your Java application, you create a Context, then ask Pnuts to load the file at the path specified into that Context. Here's what the code looks like to make that happen:

```
import pnuts.lang.*;
import java.io.*;

public class ScriptRunner {
   public static void main(String[] args) throws
IOException {
   try{
          Context context = new Context();
          Pnuts.loadFile(args[0], context);
```

KNOW WHO'S WATCHING YOU.

PROTECT YOUR PRIVACY
MONITOR COMMUNICATIONS
SECURELY ERASE FILES
BLOCK ANNOYING WEB ADS

SUPPORT FOR APPLE MAIL



Allume Systems

A Division of Smith Micro Software

Available from your favorite retailers and catalogs. Download a FREE Trial at Allume.com!

One hundred percent 10.4 "Tiger" compatible! Download a FREE trial version of Internet Cleanup at **http://www.allume.com/mac/cleanup/** today!

®, TM, and © 2005 Allume Systems, Inc. Mac OS X 10.3 "Panther" is required for Internet Cleanup 3.0.

```
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}
```

The Pnuts script to create a `JTree`, put it in a `JFrame`, and show the `JFrame` looks like this:

```
import("javax.swing.JFrame")
import("javax.swing.JTree")

frame = new JFrame();
frame.setDefaultCloseOperation(JFrame::EXIT_ON_CL
OSE)
tree = new JTree();
frame.getContentPane().add(tree);
frame.pack();
frame.show();
```

JRuby

To integrate JRuby into your Java application, you create an instance of Ruby and then ask it to load the file at the path specified:

```
import org.jruby.*;
import java.io.*;

public class ScriptRunner {
    public static void main (String[] args)
throws Exception {
        Ruby runtime =
org.jruby.Ruby.getDefaultInstance();
        try {
            File f = new File(args[0]);
            runtime.loadFile(f, false);
        } catch(Exception e) {
            System.out.println("Exception
while sourcing file " + args[0]);
            e.printStackTrace();
        }
    }
}
```

The JRuby script to create a `JTree`, put it in a `JFrame`, and show the `JFrame` looks like this:

```
require 'java'
module JavaSwing
  include_package "javax.swing"
  include_package "java.awt.event"
end

frame = JavaSwing::JFrame.new("Simple Ruby
App")
tree = JavaSwing::JTree.new()
frame.getContentPane().add(tree)
frame.setDefaultCloseOperation(JFrame::EXIT
_ON_CLOSE)
frame.pack()
frame.setVisible(true)
```

Lessons Learned

In the trivial example I used, you can see that integration is simple and that the different interpreters complete the tasks similarly. The scripts for the different languages are also similar. I think that you need to look at bigger, more complex examples than we have space for here to see the differences among the interpreters. To pick the best one for your application, you'll have to look deeper at the language syntax and feature set that the scripting interpreters support and decide for yourself if you like what you see.

In the process of doing these integrations, I took notes about what seemed most important, or interesting, or noteworthy about the different interpreters and have listed those notes below. Obviously, these thoughts represent my personal opinion. Software developers are free-thinking people, so I know you'll come to your own conclusions when you do your own due diligence.

Jacl supports the Tcl syntax, which is not difficult to learn. The Tcl programming language is already well known to many software developers, and numerous books and online tutorials are available on Tcl programming. One recommended interactive tutorial on Tcl programming is the TclTutor program, which takes you through the steps of learning to program in Tcl interactively.

JRuby supports the Ruby syntax. The RubyCentral homepage states that Ruby "combines the object-oriented power of the classic OO language Smalltalk with the expressiveness and convenience of a scripting language such as Perl." I wasn't familiar with Ruby, but the developerWorks article "Take a Shine to JRuby" (September 2004) gives you a taste of JRuby's strengths and shows a more elaborate JRuby programming example than I have in this article.

BeanShell 2.0 release notes state that BeanShell is now capable of interpreting ordinary Java source files, which is impressive. I tested this functionality and found it to work fine for the simple programs I asked it to load and run. If you want to learn more about BeanShell programming, check out the BeanShell tutorial on the BeanShell Website.

Jython supports the Python syntax. If you are unfamiliar with Python and don't have one of the Jython books handy, one place to start learning about Python is with Guido van Rossum's Python tutorial. One feature of the Python syntax is that it doesn't use braces to group statements together, it uses a combination of the colon character (:) and space indentation. This might seem like it could lead to confusion in the code, but consistent usage of spaces instead of tabs is all that's required to keep things clear. If you are looking for in-depth programming advice on Jython, several good books are available.

Rhino supports the JavaScript syntax, which is straightforward and very readable. This language has been well documented and is already known to most Web developers. That might be all the reason you need to make this the obvious choice for you. The documentation is well done. The presence of a debugger is a strong selling point for those writing complex or lengthy scripts. The Mozilla Website has a tutorial on embedding Rhino.

Q: What do all these Macintosh all-stars have in common?

Scott Kelby

Robin Williams

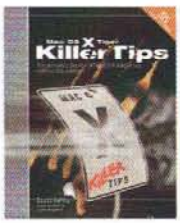
Jim Heid

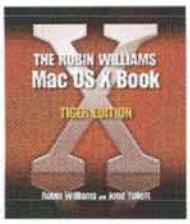
Maria Langer

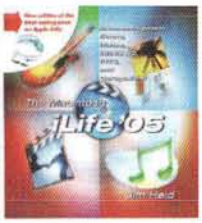
Schoun Regan

Tom Negrino

A: Peachpit!







Save 35% off the retail price, plus enjoy free domestic U.S. shipping. Visit **www.peachpit.com** and enter in coupon code **EE-FLYR-6787** when you reach the checkout page.

Offer expires 12/31/2006.

www.**peachpit**.com

JudoScript syntax is easy to pick up from the documentation and seems to include everything you need to get a complicated programming job done: threads, function pointers, and exception-handling are all supported and well-explained. To get a sense of JudoScript's strengths, read the JudoScript White Paper. For more information, the JudoScript Website offers tutorials on the JudoScript language and on how to embed JudoScript in a Java application.

Groovy syntax is Java-like and readable. One interesting feature of Groovy is that it supports closures, which let you define a piece of programming code without declaring a class or a method. You can assign that piece of programming code (the closure) to a variable if you want, pass that variable through other functions, and call that closure whenever you need by just executing the call() function against it. As I mentioned earlier, there is a JSR committee working on the language specification for Groovy, which is good. One caveat is that the Groovy syntax is still evolving, so there is a chance that Groovy scripts written today might need to be rewritten when the committee nails down the final syntax. If you are interested in reading more about the language, several good articles cover the topic, including "Groovy, Java's New Scripting Language" (O'Reilly, September 2004) and "Groovy, Scripting for Java" (Object Computing, 2003).

The only hiccup I found with the Groovy interpreter occurred when I broke an assignment statement between two lines as shown below. In this case, the interpreter seemed to ignore the part of the expression that incremented the variable by 1, like this:

```
i = i
   + 1;
```

But doing things the following way worked just fine:

```
i = i +
1;
```

I was running a beta version of 1.0, which means the parser in that build isn't totally bulletproof. It's beta, right? Groovy is getting a lot of development attention and no doubt this glitch will be addressed in a future build.

Pnuts syntax also looks similar to Java. One interesting and useful feature of the Pnuts syntax is the module concept. Reusable scripts in Pnuts can be divided into modules, which are similar to Java packages. Those modules, once loaded, can have naming conflicts or collisions resolved in a script by adding the `use(module name)` call, which tells the interpreter which module takes precedence over other modules for finding a binding for a variable or function name. The Pnuts language syntax seems easy to pick up from the documentation, and the debugger is useful.

Peace for your fingers

Speech Recognition for Mac OS X

iListen, the top-selling Speech Recognition system for Mac OS X, lets you do almost anything with your voice. Our natural speech engine and voice command & control system takes the pain out of typing.

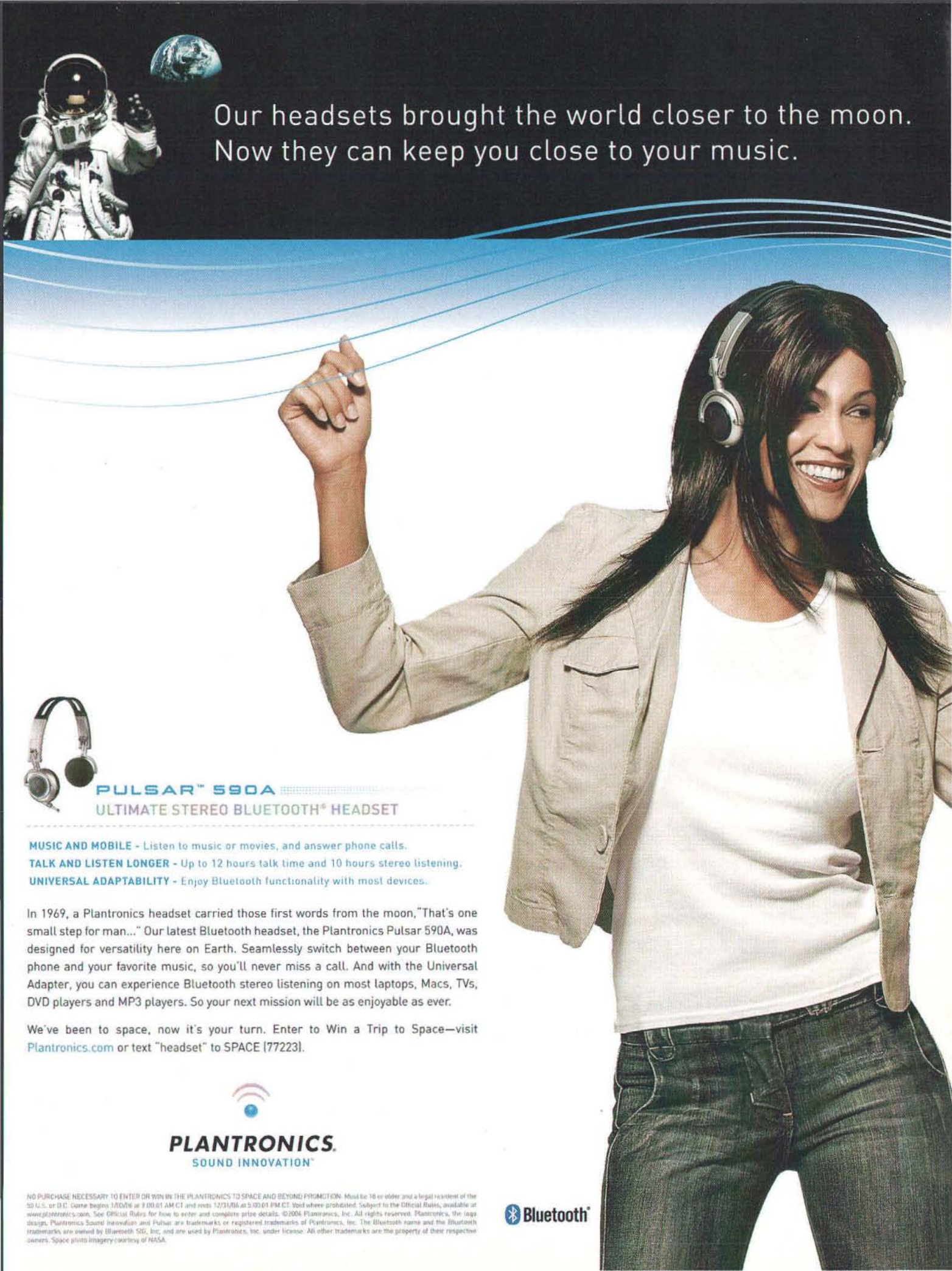
Optional unique ScriptPaks let you select all your menus by voice, there's even a ScriptPak for BBEdit, the developers choice of text and source editing.

For a limited time you can get a free BBEdit ScriptPak with any purchase of iListen from our online store. The BBEdit ScriptPak allows you to do almost anything in BBEdit with your voice that you would normally do with your keyboard and mouse. To get the free ScriptPak you must use the coupon code FREE-BBEDIT when checking out.

See www.macspeech.com/promo/bbedit.html or call (603) 251-1477 for more details.

MacSpeech

www.macspeech.com


Our headsets brought the world closer to the moon.
Now they can keep you close to your music.
PULSAR™ 590A
ULTIMATE STEREO BLUETOOTH® HEADSET
MUSIC AND MOBILE - Listen to music or movies, and answer phone calls.
TALK AND LISTEN LONGER - Up to 12 hours talk time and 10 hours stereo listening.
UNIVERSAL ADAPTABILITY - Enjoy Bluetooth functionality with most devices.
In 1969, a Plantronics headset carried those first words from the moon, "That's one small step for man..." Our latest Bluetooth headset, the Plantronics Pulsar 590A, was designed for versatility here on Earth. Seamlessly switch between your Bluetooth phone and your favorite music, so you'll never miss a call. And with the Universal Adapter, you can experience Bluetooth stereo listening on most laptops, Macs, TVs, DVD players and MP3 players. So your next mission will be as enjoyable as ever.
We've been to space, now it's your turn. Enter to Win a Trip to Space—visit Plantronics.com or text "headset" to SPACE (77223).
PLANTRONICS.
SOUND INNOVATION™
NO PURCHASE NECESSARY TO ENTER OR WIN IN THE PLANTRONICS TO SPACE AND BEYOND PROMOTION. Must be 18 or older and a legal resident of the 50 U.S. or D.C. Game begins 1/30/06 at 9:00:01 AM CT and ends 12/31/06 at 5:00:01 PM CT. Void where prohibited. Subject to the Official Rules, available at www.plantronics.com. See Official Rules for how to enter and complete prize details. ©2006 Plantronics, Inc. All rights reserved. Plantronics, the logo design, Plantronics Sound Innovation and Pulsar are trademarks or registered trademarks of Plantronics, Inc. The Bluetooth name and the Bluetooth trademarks are owned by Bluetooth SIG, Inc. and are used by Plantronics, Inc. under license. All other trademarks are the property of their respective owners. Space photo imagery courtesy of NASA.
Bluetooth®

If you are looking for more information comparing the programming interfaces of the interpreters, you might look at "Embedding APIs of Java-Based Scripting Engines" which compares the Jacl, BeanShell, Jython, Rhino, Groovy, and Pnuts interpreters performing common tasks such as evaluating an expression from a string, catching an exception thrown by a script, or setting or getting the value of a scripting variable.

The Third Benchmark: Licensing

Although these interpreters are easy to download and play with, what do their licensing rules say about you embedding it in a commercial application that you sell to customers?

The answer is that licensing is not a problem for any of these libraries. The way I read the license agreements, in each case, the user must abide by the GNU Lesser General Public license or an equivalent. This means that you can ship the libraries with your application even if your application is not free. However, you cannot strip the copyright headers out of their source files or script files and may have to clarify to users that the rights to the scripting interpreter bundled with your application belong to someone else. Of course, if you're embedding the interpreter in a commercial application, the smart thing is for you and your company attorney to carefully look at the licensing agreement.

Final Thoughts

If you need to integrate scripting support code into your Java application, my advice is to pick a single scripting interpreter and standardize on it. Costs are associated with each scripting language you support in your product, so don't make more work for yourself by trying to hook more than one scripting interpreter into your application. When adding scripting support, you can further simplify things by using an interpreter written in Java instead of a native interpreter such as Python or Tcl. That will make your solution more portable and simplify the integration task between your Java program and the interpreter.

If your developers or customers are already familiar with a particular scripting language like Tcl, Python, Ruby, or JavaScript, obviously you'll want to look seriously at the interpreter that supports that language (Jacl, Jython, JRuby, or Rhino, respectively). If you don't have that constraint, you will have a harder choice. In some ways, it's a bit like going to a new car lot. All of the choices will work, so you are left with balancing the differences between the alternatives, such as performance or options.

Some of these interpreters perform simple tasks faster than others. Some are updated and released more often, or have better documentation or debugging facilities than others. Some support compilation of scripts to bytecode. Some have language syntaxes that will either appeal to a developer or not, depending on preference, programming background, and the specific task at hand. As with most engineering tasks, you have to define your requirements and then investigate some to come up with the right answer.

If I had to distill what I learned from working with the different interpreters down to a bare minimum, here's what I'd say:

Jacl eases your entrance into scripting. Integration is simple, and if you need your scripts to be written in Tcl, it works well. If speed is your top priority, you may want to consider other choices.

Jython is one of the fastest scripting interpreters. From looking at the Website, it seems that Jython development is about to renew, which is good news. There are several good books on Jython. If you like the Python language, Jython is a solid choice.

BeanShell is not as fast as the quickest of the interpreters, but the 2.0 release supports loading of ordinary Java source, which is a strong selling point. I tried loading and running several Java source programs as scripts and found BeanShell to work fine, which is impressive. The libraries are well-organized and make integration simple. If performance is not the single most important criteria for your scripting interpreter and you want to write Java scripts, look at BeanShell.

Rhino is the winner of the performance benchmarking test and also supports Java-like syntax in its scripting. There are plenty of books on JavaScript available. Rhino appears to be well supported, and the distribution includes a useful debugger.

Pnuts is one of the fastest scripting interpreters. I am impressed with the completeness of the documentation, the simple usability of the debugger, and how straightforward it is to get things working with Pnuts. If the Pnuts syntax is a good fit for your needs, this interpreter deserves a good look.

JudoScript is in the middle of the pack for the performance benchmark, but supports a JavaScript-like syntax that is easy to learn. The documentation seems to be thorough and well organized, and the distribution includes lots of script examples. I ran version 0.9, which seemed to work well.

JRuby brings the feature set of Ruby to the table. It isn't the fastest of the interpreters, but if Ruby syntax and functionality is important to you, take a look at this interpreter. I ran version 0.8, which seemed to work just fine for my simple tests.

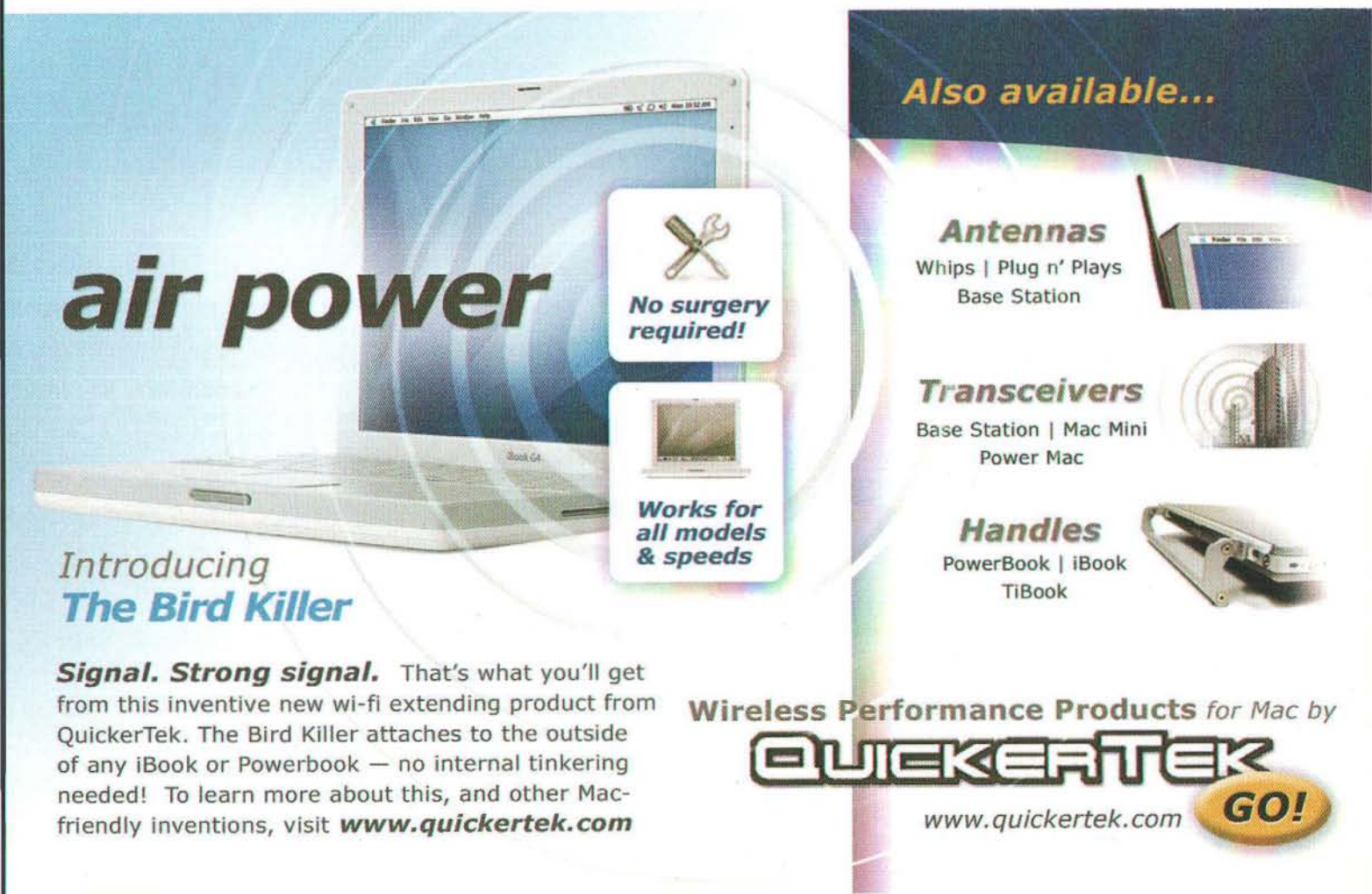
Groovy has attracted quite a bit of attention and development effort in the Java community. It is one of the fastest interpreters on the benchmarking tests, even without compiling the scripts down to classfiles. The syntax is Java-like and supports some powerful features that Java doesn't. This is an interesting addition to the programmer's toolkit that has a lot of potential.

Resources

* Download Jython:
http://www.jython.org/download.html
* "The Python Tutorial," Guido van Rossum (Python Software Foundation, 2004):
http://docs.python.org/tut/tut.html
* Download Rhino from Mozilla:
http://www.mozilla.org/rhino/download.html
* Mozilla tutorial on embedding Rhino:
http://www.mozilla.org/rhino/tutorial.html
* Download Jacl 1.3.1:
http://sourceforge.net/projects/tcljava
* Swank, the graphical toolkit for use with Jacl:
http://www.onemoonscientific.com/swank/index.html
* Download TclTutor, an interactive learning program for Tcl:
http://www.msen.com/~clif/TclTutor.html
* Download BeanShell:
http://www.beanshell.org/download.html
* BeanShell tutorial:
http://www.beanshell.org/manual/contents.html
* Download JRuby:
http://jruby.sourceforge.net
* RubyCentral, a Web resource for Ruby:
http://www.rubycentral.com
* "alt.lang.jre: Take a Shine to JRuby," Michael Squillace and Barry A. Feigenbaum (developerWorks, September 2004):
http://www-128.ibm.com/developerworks/library-combined/j-alj09084
* Download JudoScript:
http://www.judoscript.com/download.html
* JudoScript tutorial:
http://www.judoscript.com/books/judoscript-0.9/toc_details.html
* JudoScript White Paper:
http://www.judoscript.com/articles/whitepaper.html
* Tutorial on embedding JudoScript: "Embed JudoScript In Java," James Jianbo Huang (judoscript.com, December 2001):
http://www.judoscript.com/articles/embed.html
* Download Pnuts:
http://pnuts.org/snapshot/latest
* Download Groovy:
http://groovy.codehaus.org/Download
* "Groovy, Java's New Scripting Language," Ian F. Darwin (O'Reilly, September 2004):
http://www.oreilly.de/artikel/groovy.html
* "Groovy—Scripting for Java," Mark Volkmann (Object Computing, 2003):
http://www.ociweb.com/jnb/jnbFeb2004.html
* "Embedding APIs of Java-Based Scripting Engines" (pnuts.org):
http://pnuts.org/~tomatsu/embedding.html

About The Author

Paul T. Ammann has been working in IT for almost 20 years now. He is happily married to his wife Eve for 6 years. He finds writing the author's bio the toughest part of the article.



Advertiser/Product Index

4D, Inc. 53
A-1 Quality Products, Inc. 57
Aladdin Knowledge Systems, Inc. 15
Allume Systems, Inc. 11
Allume Systems, Inc. 51
Allume Systems, Inc. 65
Allume Systems, Inc. ..... BC
Andescotia LLC 9
BetterRam.com 49
Brad Sniderman 34
Brian Loomis 50
CRYPTOCard Corporation 14
Daystar Technology ..... IBC
Equilibrium 25
FairCom Corporation 1
GizMac Accessories LLC 27
GLUON/CaptureWorks 13
Harman Multimedia 19
IDG World Expo Corporation 37
IGC/MaxEMail.com 47
InterSystems Corporation 21
I.R.I.S. Group USA 2
Kerio Technologies, Inc. 44
LTA Projects 43
MacResource Computers 30
MacSpeech, Inc. 68
MacTech Cables 54
MacTech Domains 23
MacTech Printer Supplies 62
MARWARE, Inc. 57
Maxell 29
Maxum Development 66
Meta Communications 35
Mindfire Solutions 64
MYOB US, Inc. 45
Netopia, Inc. 33
OmniPilot Software, Inc. 4
Paradigma Software 18
Peachpit Press 67
Plantronics 69
Polarian Technology 63
Polycom, Inc. 55
Protective Solutions, Inc. 57
QuickerTek 71
RadTech 24
Seapine Software, Inc. ..... IFC
Small Dog Electronics 59
SpiderWorks 41
Tellurium Communications 31
Utilities4Less.com 52

Accelerators/Upgrades • Daystar Technology ..... IBC
AccountEdge • MYOB US, Inc. 45
Andescotia LLC • Andescotia 9
AquaZone • Allume Systems, Inc. 11
Bags • LTA Projects 43
c-tree Plus • FairCom Corporation 1
Cables • MacTech Cables 54
Caché • InterSystems Corporation 21
CaptureWorks • GLUON/CaptureWorks 21
CRYPTOCard Corporation 14
Digital Rights Management • Aladdin Knowledge Systems, Inc. 15
Domain Registration • MacTech Domains 23
Digital Storage Manager • Meta Communications 35
Equilibrium 25
Harman Multimedia • Harman Multimedia 19
Hosted Store • Brian Loomis 50
iKeyChain • A-1 Quality Products, Inc. 57
iListen • MacSpeech, Inc. 68
InstallerMaker, StuffIt • Allume Systems 51
Internet Cleanup • Allume Systems, Inc. 65
Kerio Mail Server • Kerio Technologies, Inc. 44
Lasso • OmniPilot Software, Inc. 4
Law Offices • Brad Sniderman 34
Long Distance Phone Service • Utilities4Less.com 52
MacResource Computers • MacResource Computers 30
MacWorld Expo • IDG World Expo Corporation 37
MarketBlast • 4D, Inc. 53
MaxEMail.com • IGC/MaxEMail.com 47
Mindfire Solutions 64
Peachpit Press • Peachpit Press 67
PhonePipe • Tellurium Communications 31
Plantronics • Plantronics 69
PowerBook Accessories • QuickerTek 71
Print Supplies • MacTech Printer Supplies 62
RadTech 24
Ram • BetterRam.com 49
Readiris Pro • I.R.I.S. Group USA 2
Rumpus • Maxum Development 66
Screen Mimic • Polarian Technology 63
Screen Protection • Protective Solutions, Inc. 57
SmallDog.com • Small Dog 59
SpiderWorks ebooks • SpiderWorks 41
Stuffit Deluxe • Allume Systems, Inc. ..... BC
Tapes • Maxell 29
Test Track Pro • Seapine Software, Inc. ..... IFC
Timbuktu • Netopia, Inc. 33
Trail Vue • MARWARE, Inc. 57
Valentina 2 • Paradigma Software 18
VoIP Phones • Polycom, Inc. 55
XrackPro • GizMac Accessories LLC 27

The index on this page is provided as a service to our readers. The publisher does not assume any liability for errors or omissions.

How to Stop Racing the Clock.

☐ Work Longer? ☐ Work Harder?

☑ Energize Your Mac!

We know your day keeps getting longer and longer. With every release of software, your Mac is bogged down even more. With every click, there's a pause. You find yourself working longer, working harder. *A faster Mac means that you can work faster, not harder - be more productive!*

Let the original Mac Performance Shop help. Daystar has been creating Mac speed for over 16 years. Whether your bottleneck is storage, connectivity or just raw CPU speed, we deliver the performance you need, where you need it.

CPU Upgrades for Raw Speed. We upgrade any Power Macintosh, any iMac Flat Panel, any PowerBook G3 and some PowerBook G4s.

Fast and Large Storage for Real-Time Video. Our **TURBO***SATA* solutions can make your drives perform like RAM. Projects open in a flash and edit in real-time.

Extreme Wireless. Wireless is great, unless you're getting slow transfers. Even Airport Extreme's are slow when the signal is weak. Daystar can boost your signals and energize your wireless network.

But, if You Really need a G5? Daystar is the only Mac Performance Manufacturer that is also an Apple Authorized Reseller. Not only can you trade-in your system for the latest and greatest... but the Daystar Pro's can upgrade it for maximum performance!

Call 877-439-8646 and beat the clock.

Authorized Reseller

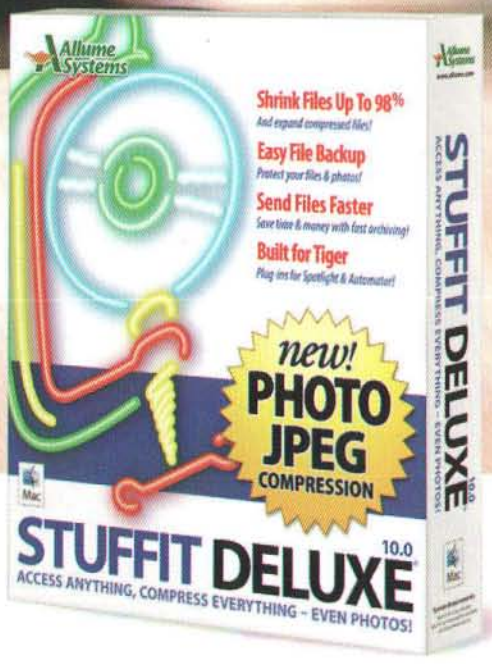
Daystar TECHNOLOGY

Daystar Technology - Your Macintosh Performance Shop
5018 Bristol Industrial Way, #202, Buford, GA 30518 USA
Toll Free: 877-439-8646 or 770-614-5400

Daystar-Tech.com **Daystar-Forum.com** **Daystar-Store.com**

THINK SMALL, THINK *FAST.*

COMPRESS JPEGS UP TO 30%
WITH NO QUALITY LOSS!
BUILT FOR MAC OS X "TIGER"
BACKUP TO .MAC, CD, & DVD



Allume Systems
A Division of Smith Micro Software

Available from your favorite retailers and catalogs. Download StuffIt Expander for FREE at www.stuffit.com!

Download a FREE trial version of StuffIt Standard Edition at **www.allume.com.** For information about StuffIt's industry-leading JPEG compression, please visit **www.stuffit.com.**

® TM, and © 2005 Allume Systems, Inc. Mac OS X 10.4 "Tiger" is required for StuffIt Deluxe's Spotlight and Automator support.